UNIVERSITY OF ALBERTA

AN OBSERVATIONAL STUDY OF THE ECLIPSING BINARY HD219634

BY

DAVID A. LYDER

A THESIS SUBMITTED TO THE FACULTY OF GRADUATE STUDIES IN PARTIAL FUFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF MASTER OF SCIENCE .

DEPARTMENT OF PHYSICS

EDMONTON,ALBERTA

(FALL, 1989)

UNIVERSITY OF ALBERTA

RELEASE FORM

NAME OF AUTHOR : DAVID A. LYDER

TITLE OF THESIS : AN OBSERVATIONAL ANALYSIS
OF THE ECLIPSING BINARY
HD219634.

DEGREE : MASTER OF SCIENCE

YEAR THIS DEGREE GRANTED : 1989

PERMISSION IS HEREBY GRANTED TO THE UNIVERSITY OF ALBERTA LIBRARY TO REPRODUCE SINGLE COPIES OF THIS THESIS AND TO LEND OR SELL SUCH COPIES FOR PRIVATE, SCHOLARLY OR SCIENTIFIC PURPOSES ONLY.

THE AUTHOR RESERVES OTHER PUBLICATION RIGHTS, AND NEITHER THE THESIS NOR EXTENSIVE EXTRACTS FROM IT MAY BE PRINTED OR OTHERWISE REPRODUCED WITHOUT THE AUTHOR'S WRITTEN PERMISSION.

David A Lyder
(Student's Signature)

APT. 170 8745-165 St
(Student's Permanent Address)
EDMONTON, AB

Date : September 18, 1989.

THE UNIVERSITY OF ALBERTA

FACULTY OF GRADUATE STUDIES AND RESEARCH

The undersigned certify that they have read, and recommend to the Faculty of Graduate Studies and Research, for acceptance, a thesis entitled **"AN OBSERVATIONAL STUDY OF THE ECLIPSING BINARY HD219634"** submitted by **DAVID A. LYDER** in partial fulfillment of the requirements for the degree of Master of Science.

SUPERVISOR

Wytze Brouwer

EXTERNAL EXAMINER

Date: September 8, 1989

Dedication

To my parents, Nuala and Brian Lyder. Thank you for being there when I needed you.

ABSTRACT

This thesis presents the photometric (B and V filters) analysis of the eclipsing binary HD219634 in detail. The first part presents the modern description of binary systems in terms of the Roche lobe potentials. This is followed by a description of the least-squares fitting to the light curves using the Wilson-Devinney code. The code has been adapted to run in an AT environment. Difficulties in using the code are discussed in detail. Finally, the solution and possible models for HD219634 as suggested by the photometry are discussed. An annotated listing of the Wilson-Devinney code, including modifications, and original computer programs are presented in the appendices.

Acknowledgements

During the preparation and writing of this thesis many people have helped with words of encouragement and assistance. In particular I would like to thank Dr. Douglas Hube for showing confidence in me and allowing me the opportunity to study astronomy. I am most heavily indebted to Brian Martin for his patience and understanding in showing me how to walk along the road of Science. I would also like to thank Joan Hube and my committee members, particularly Dr. W. Brouwer for their helpful comments and encouragements. Finally, I would like to thank Ruth Butler for her patience, understanding and help in preparing this thesis.

Table of Contents

Chapter Page

1 Chapter One 1

1.1. Introduction 1

2 Classical Method of Spectroscopic Analysis 5

2.1. Introduction 5

2.2. Physical Principles 5

2.3. Method of Lehmann-Filhes 7

2.3.1. Differential Corrections 9

2.4. Difficulties 12

3 Classical Methods of Photometric Analysis 15

3.1. Introduction 15

3.2. Russell's Method 16

3.2.1. Underlying Principles 16

3.2.2. Total Eclipses 18

3.2.3. Partial Eclipses 19

3.2.4. Rectification 21

Chapter .... Page

3.3. Kopal's Method 23

3.3.1. Underlying Principles 23

3.3.2. Roche Model 24

3.4. Physical Principles of Light Variation 26

3.4.1. Limb-Darkening 30

3.4.2. Gravity-Darkening 31

3.4.3. Reflection Effect 37

3.4.4. Rectification 39

3.5. Conclusion 40

4 The Wilson-Devinney Approach 41

4.1. Introduction 41

4.2. Method of Analysis 41

4.2.1. Calculation of G, Gravity-Darkening 45

4.2.2. Calculation of R, Reflection Effect 48

4.2.3. Calculation of Limb-Darkening 50

4.3. Method of Solution 51

Chapter . . . Page

4.3.1. Matrix Methods . . . 52

4.3.2. Convergence . . . 53

4.4. Conclusion . . . 55

5 Using the WD Code . . . 56

5.1. Introduction . . . 56

5.2. Synthetic Light Curve (LC) . . . 56

5.2.1. Varying the Initial Model Parameters . . . 57

5.2.2. Checking the Solution . . . 58

5.3. The Differential Corrections Program (DC) . . . 60

5.3.1. Method of Subsets . . . 61

5.3.2. Step Sizes . . . 62

5.3.3. Input Cards . . . 63

5.4. Conclusion . . . 63

6 MR Cygni - A Test Case . . . 78

6.1. Introduction . . . 78

6.2. Objectives . . . 78

Chapter . . . . Page

6.3. The Initial Parameter Set 79

6.4. LC Results 84

6.5. DC Results 92

6.6. Conclusion 98

7 Analysis of HD219634 99

7.1. Introduction 99

7.2. Photometric Data 99

7.3. Sources of Error 101

7.3.1. Systematic Sources of Error 103

7.3.2. Random Sources of Error 104

7.3.3. Estimation of Total Error 106

7.4. Preparation for the WD Code 106

7.4.1. The Initial Model Parameters 107

7.4.2. Trial One 112

7.4.3. Trial Two 114

7.4.4. Trial Three 119

Chapter Page

7.5. Interpretation of the Results 124

7.6. Conclusion 130

8 Conclusion 133

Bibliography 135

Appendix A The Wilson-Devinney Code 139

A.1. Subroutine LC 139

A.2. Subroutine DC 150

A.3. Subroutine ATM 171

A.4. Subroutine BB1 171

A.5. Subroutine BB2 171

A.6. Subroutine BOLO 172

A.7. Subroutine CALC1 172

A.8. Subroutine DCDRIVE 173

A.9. Subroutine DGMPRD 174

A.10. Subroutine DMINV 175

A.11. Subroutine DURA 175

Chapter .... Page

A.12. Subroutine ELLONE 175

A.13. Subroutine FOUR 175

A.14. Subroutine INOUT1 175

A.15. Subroutine INPUT1 177

A.16. Subroutine INPUT2 178

A.17. Subroutine ITER 180

A.18. Subroutine KEPLER 185

A.19. Subroutine LCDRIVE 186

A.20. Subroutine LCR 186

A.21. Subroutine LIGHT 186

A.22. Subroutine LUM 187

A.23. Subroutine LUMP 187

A.24. Subroutine MATFOR 187

A.25. Subroutine MODLOG 188

A.26. Subroutine NEKMIN 189

A.27. Subroutine RING 189

Chapter Page

A.28. Subroutine ROMQSP 189

A.29. Subroutine SCRATCH 189

A.30. Subroutine SINCOS 191

A.31. Subroutine SORTER 192

A.32. Subroutine SPOT 192

A.33. Subroutine SQUARE 192

A.34. Subroutine SURFAS 192

A.35. Subroutine VOLUME 192

Appendix B Original Programs 193

B.1. Phaser 193

B.2. Gaincorr 198

B.3. SortAve 208

List of Figures

Figure . . . . Page

2.1. Simple Illustration of the Geometry of Binary System. 6

2.2. Sample Velocity Curve - Single-Lined System. 8

2.3. A Double-Lined System. 11

2.4. The 'Rotation Effect' 13

3.1. Geometric Relationship for Russell Method 17

3.1. Standard Points in the Russell Method. Total Eclipse. 20

Russell's $\Psi(k,\alpha)$ Functions. 20

3.3. Standard Points in the Russell Method. Partial Eclipse. 22

Depth Relation, χ in Russell Model. 22

3.4. A Simple Schematic of the Three Principle Types . . . .

of Close Binary Systems.. 27

3.4. Simple Illustration of Limb-Darkening. 33

3.5. A Simple Illustration of Gravity-Darkening. 36

4.1. Illustration of Stars in WD Code. 43

4.2. An Illustration of Some of the Geometry of ΔF_l. 44

Figure .... Page

4.3. The Relationship between I and cosβ. 46

5.1. A Simple Schematic of the Interdependence of some of the Parameters in the WD Code.. 59

6.1. MR Cygni - V Photometry 81

Wilson's Data 81

Author's Data 82

6.2. LC Solutions (LC1-11) 84

Estimated Data 85

DC Solutions (DC1-4,DC12-16) 94

7.1. Photometry for HD219634 102

7.2. Starting Solution for HD219634 as Suggested by LC 111

7.3. End of Fourth Iteration (Trial One) 113

7.4. Starting Solution for HD219634 (Trial Two) 115

Figure .... Page

7.5. End of Seventh Iteration (Trial Two) 118

7.6. Final Solution for HD219634 (V Photometry). 123

7.7. Final Solution for HD219634 (Coarser Scale) 125

7.8. Representations of HD219634 127

7.9 Final Solution for HD219634 (B Photometry) 132

List of Tables

Table . . . . Page

I Linear Limb-Darkening Coefficients 32

II LC Input Parameters 64

III DC Input Parameters 72

IV Starting Parameters for MR Cygni 80

V Spetroscopic Data for HD219634 108

VI Starting Parameters for HD219634 (Trial One) 110

VII Corrections for Trial Two 117

VIII Starting Parameters for HD219634 (Trial Three) 120

IX Corrections, Probable Errors, and Residuals for Third Trial. . 121

X Final Parameters for HD219634 (V Photometry) 122

XI Absolute Parameters for Semi-Detached Eclipsing Binaries 129

XII Final Parameters for HD219634 (B Photometry) 131

CHAPTER ONE

1.1. Introduction

Binary stars serve as a fundamental source for knowledge of key stellar evolutionary parameters such as mass and radius. A binary system is a gravitationally bound system consisting of two or more stars whose relative motion may be described by Kepler's laws. As such, an accurate description of the orbit yields accurate information about the masses of the components. In addition, a description of the light variation during a cycle gives information regarding the physical state and relative dimension of the components.

Classical methods of analysis relied on point mass approximations. For close binaries, systems with short periods, there will be proximity effects e.g. gravity-darkening and the reflection effect which must be properly considered. With the advent of modern computing techniques, it has become possible to analyze binary systems using sophisticated modelling which is physically realistic. This thesis presents the analysis of an eclipsing binary system, HD219634, using the Wilson-Devinney code (Wilson and Devinney, 1973), hereafter referred to as the WD code, with a discussion of the results in terms of current theories of stellar structure and stellar evolution.

HD219634 is an early B-type system (B0 - B3) and, as Popper (1980) has noted, binaries with components earlier than B6 do not,in general, have well-resolved spectra. Use was made of advanced cross-correlation techniques in the analysis of the spectroscopic data for HD219634 (Gulliver, Hube and Hill, 1985).

Given that this system comprises relatively massive components and the time spent on the main sequence is inversely proportional to the mass then it should be no surprise that HD219634 is one of perhaps only half-a-dozen such systems known. In addition, this star is a double-lined spectroscopic system and in principle it is possible to obtain the masses of the components separately rather than just the mass ratio. Thus, the system is important to our understanding of the structure and evolution of early type stars.

To obtain complete physical descriptions for eclipsing binary systems it is necessary to obtain the photometry. Early type systems often demonstrate complicated photometric behaviour due to enhanced stellar winds; remnants of the stellar formation process - either gas clouds or dust grains, or a combination of the above processes (Piotrowski et al., 1974). These phenomena are further complicated as they may be transitory in nature, thus presenting slightly different configurations to an observer at different epochs. Furthermore, a close binary like HD219634 will have its photometry further complicated by the previously mentioned proximity effects. A successful model of HD219634 should take into consideration these complications.

As a major component of this thesis the photometric data were gathered for HD219634 during the period September, 1987 to December, 1988. The process of collection followed a standard procedure; observation via a two-channel differential photometry program, correction for sky conditions and the sensitivity of the two channels and, finally, transformation to a standard system.

Analysis of the data, both photometric and spectroscopic, by the WD code was consistent with a general appeal made at the IAU Colloquium 107 in the Summer of 1988. It is the opinion of many stellar astronomers that this particular code makes use of the best physics available in our modelling of binary systems; a realistic geometry which follows from a sound physical principle, the Roche potentials. Furthermore, this code employs a least-squares procedure to iteratively make a best-fit to both the spectroscopy and photometry.

The first part of this thesis deals with a discussion of some of the classical techniques for analysis of eclipsing binaries with particular emphasis on their advantages and weaknesses. A brief discussion of the WD code will follow, detailing some of the theoretical considerations and methods used in the code. There will be an example of a binary system , MR Cygni, illustrating some of the difficulties in using the WD code. Finally, a detailed analysis of HD219634 will follow. An appendix is provided with original programs used in the analysis as well as a documented and annotated WD code. The version of the code used on the AT does differ significantly from the original. With the dicussion provided in the thesis as well as the annotations future users should be succesful in using the WD code in their analysis of binary systems.

CHAPTER TWO

A Classical Method of Spectroscopic Analysis

2.1 Introduction

In this chapter we consider one of the classical techniques used in the radial velocity analysis, the method of Lehmann-Filhes. This technique is used in the WD code as its application is straightforward and differential corrections may be calculated for improved solutions.

2.2. Physical Principles

For brevity, let us consider a system where only the brightest component is visible (Fig.2.1), a single-lined system (SB1). An observer of this system would record the radial or line-of-sight velocity. This may be found by measuring the spectral line displacement as described by the classical Doppler formula

$$v_m/c = \Delta\lambda/\lambda \qquad (2.1)$$

v_m, the measured radial velocity is corrected for the diurnal and annual motion of the Earth to give the heliocentric radial velocity, v_r. The resulting heliocentric radial velocity is plotted as a function of time.

Figure withheld due to copyright restriction.

2.3. Method of Lehmann-Filhes

The relevant geometry is illustrated in Fig.2.1 and a simple schematic radial velocity curve is presented in Fig.2.2. (cf. Binnendijk (1960), p.352-357). Several simple expressions can be derived which relate orbital parameters to measurable features of the velocity curve:

$$v_r = v_0 + K\{e\cos\omega + \cos(v + \omega)\} \qquad (2.2)$$

$$A = K(1 + e\cos\omega) \qquad (2.3)$$

$$B = K(1 - e\cos\omega) \qquad (2.4)$$

thus,

$$K = 1/2\,(A + B) \qquad (2.5)$$

where A and B are the maximum displacements from v_0, the systemic velocity, and K is the semi-amplitude for the system. ω is the longitude of periastron, e is the eccentricity and v is the true anomaly. Further analysis of the system will also lead to

$$a\sin i = ((A + B)/4\pi)P(1 - e^2)^{1/2} \qquad (2.6)$$

where a is the semi-major axis of the brighter star, i the inclination of the orbit and P, the period. Units of A,B and K are km/sec, a is in km, and P in seconds.

At this point we may note it is impossible to further reduce (2.6) and obtain a and i separately. This requires further independent information from photometry. Recalling Kepler's third law,

Figure withheld due to copyright restriction.

$$P^2(M_1+M_2) = (4\pi^2/G)(a_1+a_2)^3 \tag{2.7}$$

it may be shown that in combination with (2.6) one obtains the mass function

$$f(M) = (M_2 \sin i)^3/(M_1+M_2)^2 \tag{2.8}$$

The mass function is useful in placing limits on the properties of the unseen secondary component.

2.3.1. Differential Corrections

A further advantage of this method, in addition to its simplicity is that it can be extended to permit one to calculate improved orbital elements and estimates of their uncertainties. To do this one calculates for each time of observation an expected radial velocity given the initial set of orbital parameters. To calculate a radial velocity, at a particular time,t, we make use of Kepler's equation:

$$M = E - e\sin E = 2\pi/P(t - T) \tag{2.9}$$

where M is the mean anomaly, E the eccentric anomaly, T the time of periastron passage and t the time of observation. The relationship between v and E is given by

$$\tan(v/2) = ((1+e)/(1-e))^{1/2}\tan(E/2) \tag{2.10}$$

Returning to our radial velocity equation

$$v_r = v_0 + Ke\cos\omega + K\cos(v+\omega) \tag{2.2}$$

we have an equation of condition

$$\Delta v = (\partial f/\partial v_0)\Delta v_0 + (\partial f/\partial K)\Delta K + (\partial f/\partial e)\Delta e + \ldots \tag{2.11}$$

(Petrie, p.566) which defines the change in radial velocity, produced by the corresponding changes in elements Δv_0, ΔK, etc. Δv is the measure of the residual between observed and calculated velocity or 'O - C'. For each data point or more commonly a weighted normal point one calculates the coefficients (derivatives) in (2.11) making use of (2.2), (2.9) and (2.10). Thus, one solves a system of equations to obtain the corrections or deltas. These values are then added to the original parameters to obtain the next solution. The process is repeated until the (O - C)'s are within an allowed tolerance.

A double-lined system (SB2), where both components are of comparable brightness, leads to two velocity curves as illustrated in Fig. 2.3.

For this type of system one obtains

$$(M_1+M_2)\sin^3 i = a^3 \sin^3 i / P^2 \tag{2.12}$$

with the additional information that the ratio of the semi-amplitudes is inversely proportional to the mass ratio (Aitken, p.203)

$$K_1/K_2 = M_2/M_1 \tag{2.13}$$

But again we are unable to further reduce the problem without photometric data.

Figure withheld due to copyright restriction.

2.4. Difficulties

There are several sources of error inherent in the spectroscopic study of binary systems. These include the reflection effect, rotational effect, and gas stream effect to name but a few. The first of these, the 'reflection effect', is due to the proximity of the stars. Radiation from the hotter star will be absorbed by the cooler star's facing hemisphere making it brighter than it would otherwise be.Thus, the distribution of luminosity in the system will be shifted away from the stars towards the centre of mass of the system. The net effect on the spectroscopic data will be an underestimation of a or asini.

A second complication in the spectroscopy occurs for eclipsing systems and is due to the rotation of the two stars. During ingress the crescent of the eclipsed star will show a rotational motion away from the observer. At egress, the rotational motion will be towards the observer.

At other phases the velocity is, in effect, an average over one complete hemisphere. In principle, this effect should be enhanced if the star is a rapid rotator and if the eclipse is total (Fig. 2.4).

Figure withheld due to copyright restriction.

Gas streams, the third effect, are by far the most difficult to recognize and to incorporate into a final model. As gas streams from one star to the other, lines may be blended producing distortions in the radial velocity curve. This distortion will be highly dependent on geometry, temperature and composition of the stream.

CHAPTER THREE

Classical Methods of Photometric Analysis

3.1. Introduction

As demonstrated in the previous chapter, spectroscopy alone does not provide sufficient information to obtain a complete and unique solution for binary systems. If the system happens to be eclipsing it may be possible to extract further information from photometric variation. The additional information to be obtained includes the inclination of orbit, radii of the components, and ratio of surface temperatures.

Classically, this analysis proceeds according to one of two schools of thought - the Russell method (also known as the Russell-Merrill method) or Kopal's method. The first method is a purely geometric model while the second is based on a dynamical model (the Roche model). There is a compromise in the two methods between mathematical simplicity and more representative physics. Ultimately both methods are unsatisfactory as they include the dubious technique of rectification.

3.2. Russell's Method

3.2.1. Underlying Principles

H.N.Russell in 1912 developed the theoretical principles for this method. It was a purely geometric model with several basic assumptions as its starting point:

1) Spherical stars.

2) Circular orbits.

3) Simple systems i.e. no distortions due to gas streams, flares, etc.

4) Uniform distribution of light over a star's surface.

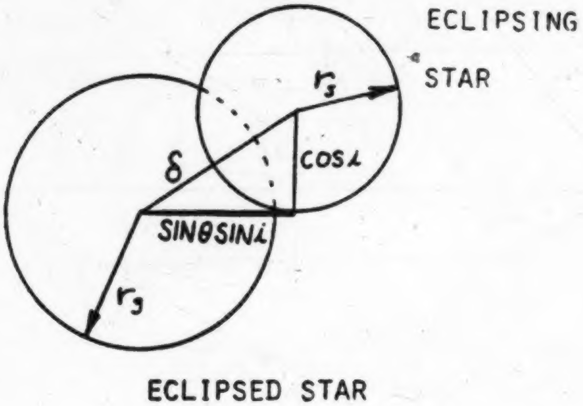
5) Symmetric eclipses.

In general, one considered two possible eclipse scenarios (see Fig.3.1.),

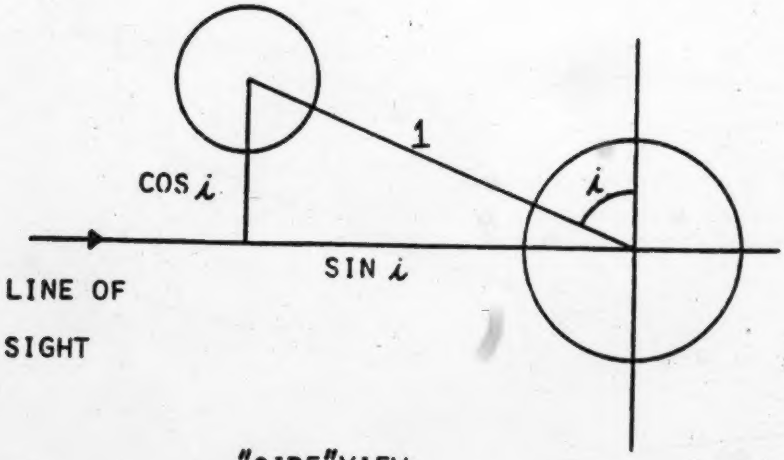
$$\left.\begin{array}{l}\text{occultation} : l_0 = 1 - l_s = L_g \\ \\ \text{transit} : L_t = 1 - k^2 L_g\end{array}\right\} \quad \text{Total (3.1)}$$

$$\left.\begin{array}{l}\text{occultation} : 1 - L_{po} = \alpha_o L_s \\ \\ \text{transit} : l_{pt} = 1 - k^2 \alpha_o L_g\end{array}\right\} \quad \text{Partial(3.2)}$$

Fig.3.1. Geometric Relationship for Russell Method (Adapted from Holmgren, p.11).



"FRONT" VIEW



"SIDE"VIEW

where L_s,L_g are the intensities of the smaller and larger stars respectively, k is the ratio of radii r_s/r_g, and the various l's are measures of the intensities throughout the corresponding event. With these relations, it is possible to generate a set of alpha-functions such that

$$\alpha/\alpha_o = (1 - l)/(1 - l_o) \tag{3.3}$$

is a measure of the intensity for a given phase relative to a_o.

In addition to (3.3) there is a parameterization given by

$$\delta^2 = \cos^2 i + \sin^2 i \sin^2 \theta \tag{3.4}$$

where δ is the distance between centres of the two components, i is the inclination and θ is the phase angle.

3.2.1. Total Eclipses

By considering three standard points within an eclipse it is possible to generate a function

$$(\sin^2 \theta - \sin^2 \theta_2)/(\sin^2 \theta_2 - \sin^2 \theta_3) = \psi(k,\alpha) \tag{3.5}$$

By definition we allow

$$\sin^2 \theta = A' \tag{3.6}$$

$$\sin^2 \theta_2 - \sin^2 \theta_3 = B' \tag{3.7}$$

with θ as some arbitrary sliding point within an eclipse.

For the standard point (α_2,θ_2) we set $\psi(k,\alpha_2) = 0$; for (α_3,θ_3), $\psi(k,\alpha_3) = -1$. Furthermore, with $\theta = 0$, $\psi(k, \alpha_2) = -A'/B'$ and thus, from

$$\sin^2\theta = A' + B'\psi(k,\alpha) \tag{3.8}$$

it is possible to plot Russell's $\psi(k,\alpha)$ functions (Fig.3.2).

By considering a set of points within $0 < \alpha < 0.4$ it is possible to obtain a set of $\psi(k,\alpha)$'s and hence, a set of k's which may then be averaged for a best k.

To solve for r_g and i we consider points of internal and external tangency (contact)

$$\cos^2 i + \sin^2 i \cos^2\theta_e = r_g(1 + k)^2 \tag{3.9}$$

$$\cos^2 i + \sin^2 i \sin^2\theta_\iota = \rho_g^2(1 - k)^2 \tag{3.10}$$

Solving (3.9) and (3.10) yields

$$r_g^2/\sin^2 i = B'/P1(k) \tag{3.11}$$

$$\cot^2 i = B'/P2(k) - A' \tag{3.12}$$

where P1(k) and P2(k) are tabulated functions.

3.2.3. Partial Eclipses

Partial eclipses are a more complicated matter. In this case, the defining equation is given by

$$\sin^2\theta(n) = A' + B'P(k,n,\alpha_0) \tag{3.13}$$

Figure withheld due to copyright restriction.

where now we must solve for k,i,r_g, and α_0 (n is simply a sliding standard point). To do this, it is necessary to first examine a 'shape relation' χ between k and a_0. This will allow a reduction of the order of the problem to that of total eclipses where k,i, and r_g are found (see Fig.3.3).

3.2.4. Rectification

Obviously there are problems with applying the above method to many binaries. Systems with short periods (P $\leq$ 3 days) will suffer from complicated phenomena which will affect the photometry. In Russell's method it is necessary to reduce the actual system to an equivalent system of spherical stars through rectification of the light curve by:

i) Removing the effects due to oblateness and reflection through normalization of the out-of-eclipse variation. With the intrinsic light given by

$$L = (A_0 - C_0) - (A_2 + C_2)\cos^2\theta \tag{3.14}$$

then rectification for reflection is given by

$$l_{r'} = (A_0 + C_0) - (A_2 - C_2)\cos^2\theta \tag{3.15}$$

followed by that for oblateness

$$l_{r''} = l_{r'}/((A_0 + C_0) - (A_2 - C_2)\cos^2\theta), \quad l_{r''max} = 1 \tag{3.16}$$

Determination of the constants is made at various out-of-eclipse phases of the light curve.

Figure withheld due to copyright restriction.

ii) Removing the effects of limb- and gravity-darkening on the eclipse variation. This is done with the use of extensive tables for theoretical values of limb-darkening coefficients and differing gravities.

It is obvious that this method has very limited applications as it suffers from two basic weaknesses:

i) Modelling of the more complicated effects, particularly oblateness and reflection is wrong.

ii) The equivalence between a spherical star and a distorted star does not hold. A proper physical description of the two situations is completely different.

3.3. Kopal's Method

3.3.1. Underlying Principles

Short period binary stars are not normally spherical but are distorted due to axial rotation and tidal distortion, as has been discussed above. A generalized representation for a component of such a system is given by

$$r = a\{1 + Y_{rot} + \Sigma\, Y_j\} \quad j=2,3,4 \tag{3.17}$$

where Y_j is the jth surface harmonic representing the tidal distortion caused by its companion.

This representation is based upon the following assumptions:

i) Periods of free oscillation are much shorter than the orbital period. As a natural consequence of this assumption it is found that distortion is a tidal phenomenon. Furthermore, level-surfaces of constant density coincide with those of constant potential.

ii) A component's mass is centrally condensed. If a star becomes highly distended during its evolution and a substantial fraction of its mass is in its atmosphere this assumption is not valid.

iii) Second and higher order contributions of axial rotation are ignored.

3.3.2. Roche Model

To develop this model one now considers two stars of masses m_1 and m_2. If we place the origin of our coordinate system at m_1 and set our x-axis to coincide with the line joining the centres of the two stars, while the z-axis is perpendicular to the plane of orbit then it follows that the potential Ψ at a point p(x,y,z) is given by

$$\Psi = Gm_1/r + Gm_2/r' + (w^2/2)\{(x - m_2R/m_1+m_2)^2 + y^2\} \qquad (3.18)$$

where R is the separation of the components and

$$\left.\begin{aligned} r^2 &= x^2 + y^2 + z^2 \\ r'^2 &= (R - x)^2 + y^2 + z^2 \end{aligned}\right\} \qquad (3.19)$$

with

$$\omega^2 = G(m_1+m_2)/R^3 \tag{3.20}$$

defining the Keplerian angular frequency.

Switching to spherical coordinates with R set to unity it follows that

$$\left.\begin{aligned} x &= r\cos\phi\,\sin\theta = r\lambda \\ y &= r\sin\phi\,\sin\theta = r\mu \\ z &= r\cos\theta = r\nu \end{aligned}\right\} \tag{3.21}$$

and thus, our potential is now given by

$$\begin{aligned} \Omega &= 1/r+q\{1/(1-2\lambda r+r^2)^{1/2} - \lambda r\}+((q+1)/2)r^2(1-\nu^2) \\ &= (R\Psi/Gm) - (m'^2/2m(m+m')) \end{aligned} \tag{3.22}$$

with $q = m_2/m_1$.

It is a version of (3.22) of the potential (and derivatives thereof) which is used in the WD code. We note that for large Ω, the stars occupy separate ovals with small r and r'.

If, on the other hand, Ω is small than it is possible that a star may extend to the largest closed equipotential (Roche limit) capable of containing its entire mass. When this occurs for one star the system is said to be semi-detached; when both stars reach their respective limits we have a contact system. If one component 'attempts' to expand beyond this limit, mass overflow occurs initially through the inner Lagrangian point, L_1. It may be shown (Kopal (1959), p.134) that the distance along the x-axis to the point of overflow for a star centred at (0,0,0) is given by

$$x_1 = 1 + w + 1/3\, w^2 + 1/9\, w^3 + \ldots \tag{3.23}$$

where $w^3 = q/(1 + q)$. To obtain x_1 for the companion set $q = 1/q$ and $x_1 = 1 - x_1$.

Kopal's description of a star's shape, radius, etc. via the Roche potential surfaces is certainly an improvement over Russell's spherical stars (see Fig.3.4). Furthermore, his treatment of the underlying physical processes governing light variation during a binary's cycle i.e. gravity- and limb-darkening, the reflection effect , etc., is more physically realistic.

3.4. Physical Principles of Light Variation.

To calculate the light variation from a binary system requires that one considers radiant energy flow through stellar atmospheres, and the spectral and spatial energy distribution over the surface. This description is based upon two physical principles:

i) From conservation of energy one may obtain the distribution of brightness over a star's surface via an equation of transfer given by

$$dI/dS = \kappa\rho(B - I) \tag{3.24}$$

(Kopal (1959), p.150) with κ defined as the opacity, ρ the density, B the emissivity of the source material (source function) and I the intensity of the source. S is the distance light must travel through the star's atmosphere, thus

$$d/ds = \mu(d/dr) + (1 - \mu^2 - \nu^2)/r(d/d\mu) + \mu\nu/r(d/d\nu) \tag{3.25}$$

Figure withheld due to copyright restriction.

with $\mu = \cos\theta$, $\nu = \cos\phi \sin\theta$. By assuming that the depth at the boundary for which the star is semi-transparent is much less than the total radius then our model is reduced to plane-parallel layers.Hence, (3.24) reduces to

$$\mu dI/d\tau = I - B \tag{3.26}$$

with the optical depth τ defined by

$$d\tau = -\kappa\rho dr \tag{3.27}$$

We also allow orientation of our reference frame such that

$$\mu = \cos\gamma \tag{3.28}$$

with γ being the angle between the line joining our two stars and our line of sight.

ii) We assume that our stars are in radiative equilibrium i.e. there are no sources or sinks in the atmosphere so that our parallel layers transmit, without gain or loss, energy which originated at the core. This allows us to determine a relationship between B and I.

With B representing the radiation emitted at any particular point one obtains

$$B(\tau) = 1/2\int_0^\pi I\sin\gamma\, d\gamma = 1/2\int_{-1}^{1} I\, d\mu \tag{3.29}$$

thus, (3.25) reduces to

$$dI/d\tau = I - 1/2\int_{-1}^{1} I\, d\mu \tag{3.30}$$

(Bowers and Demming p.79) equation (3.30) describes the transfer of energy through the plane-parallel atmosphere. The intensity of light $I(\tau,u)$ at any optical depth τ in an arbitrary direction u will be given by the solution of (3.30). As boundary conditions for this problem we have as the net flux, F given by

$$F = 2\int_{0}^{\pi} I \sin\gamma \cos\gamma\, d\gamma = 2\int_{-1}^{1} I\, \mu d\mu \tag{3.31}$$

F must be constant and independent of τ. We also have

$$I(0,\mu) = 0, \quad -1 < \mu < 0 \tag{3.32}$$

With these constraints it follows that the normalized distribution of surface brightness $J(\mu)$ between the centre and the limb is given by

$$I(0,\mu) = I(0,1)\{I(0,\mu)/I(0,1)\} = HJ(\mu) \tag{3.33}$$

with H being the intensity of radiation emerging normally to the surface, $I(0,1)$.

So far our discussion has dealt with the total flow of radiant energy emerging from a star's atmosphere. However, we normally only gather photometric data at a particular wavelength, λ. To obtain the distribution of brightness at a particular λ we recall Planck's law

$$B_\lambda(T) = (2hc/\lambda^3)/(\exp(hc/\lambda kT) - 1) \qquad (3.34)$$

The intensity of emergent radiation is given by

$$I_\lambda(0,\mu) = \int_0^\infty B_\lambda(\tau) \exp((-\kappa_\lambda/\underline{\kappa})\sec\gamma)(\kappa_\lambda/\underline{\kappa}) \sec\gamma \, d\tau \qquad (3.35)$$

Solutions for (3.35) are found by expanding (3.34) around a point $\tau = \tau_{eff}$ at which the actual temperature, T in the atmosphere is equal to the effective temperature of the star[1].

3.4.1. Limb-Darkening

Finally, the general solution is given by

$$I_\lambda(0,\mu) = H(1 - \mu_1 - \mu_2 - \ldots + \mu_1\cos\gamma + \mu_2\cos\gamma + \ldots) \qquad (3.36)$$

which is the law of limb-darkening. In most cases we only consider first order terms

$$I_\lambda(0,\mu) = H(1 - \mu_1 + \mu_1\cos\gamma) \qquad (3.37)$$

Limb-darkening coefficients have been calculated by Kopal (1959) as a function of T_{eff} and spectral type. Sample values are presented in Table I (see also Fig.3.5.).

Limb-darkening tends to darken those visible surfaces most distant from the observer. The values listed in Table 1 are theoretical and should be accepted with some caution. For earlier spectral types (B8-A2) 'line blanketing' at longer λ's, above the Balmer limit, tends to reduce the effects of limb-darkening. Stars later than F2 show this throughout the blue-violet range (Kopal(1959),p.160-61).

3.4.2. Gravity-Darkening

Stars which are distorted will be affected by an additional phenomenon, namely, gravity-darkening, which will alter the distribution of brightness over their surfaces. This effect, first discussed by H. von Zeipel (1924) was described as follows

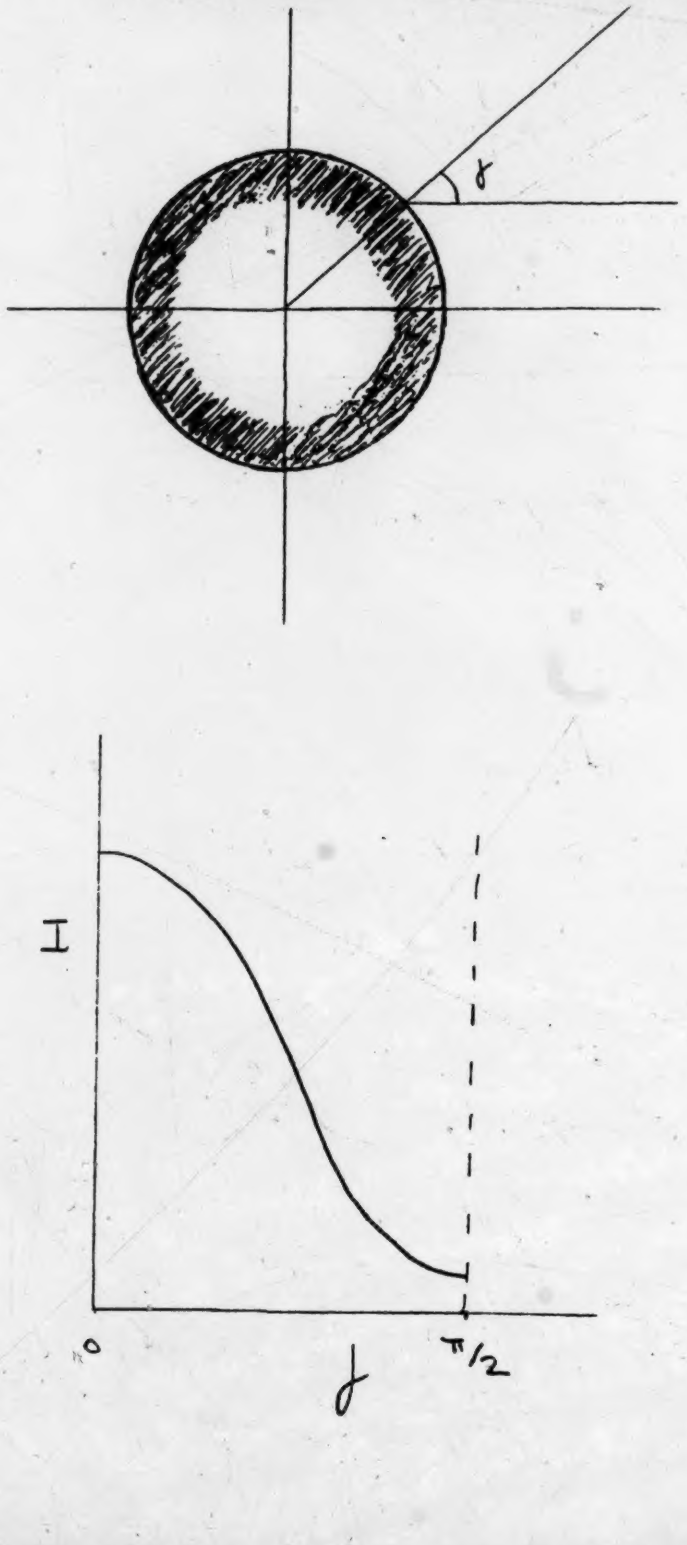
> "The emergent flux of total radiation over the surface of a rotationally or tidally distorted star in radiative equlibrium varies proportionally to the local gravity."(Kopal, 1959. p.170).

Let us consider the flux, F, emerging normally to the surface of a star. For a star with a purely radiative atmosphere we have

$$F = -(c/\kappa\rho)\mathrm{grad}P_r \qquad (3.38)$$

Figure withheld due to copyright restriction.

Fig.3.5. Simple illustration of Limb-darkening.



(Kopal (1978), p.192-193) where κ is the optical opacity[2], ρ is the local density, c is the speed of light and P_r is the radiation pressure. grad, the gradient operator reduces from symmetry to (d/dn). Hence,

$$F = -(c/\kappa\rho)(dP_r/dn) \tag{3.39}$$

Obviously, the larger the pressure gradient the greater the flux of the photons through a unit surface area. We also have from thermodynamics

$$P_r = (1/3)aT^4 \tag{3.40}$$

where a is the radiation constant. Thus,

$$\begin{aligned} F &= -(c/\kappa\rho)(d/dn((1/3)aT^4)) \\ &= -(4ac/3\kappa\rho)T^3 dT/dn \end{aligned} \tag{3.41}$$

T here is the local thermodynamic temperature. The coefficient of (3.41) is coefficient of radiative heat conduction. It remains then to relate this expression to the equation of state defining our star. This follows naturally from our assumption of hydrostatic equilibrium

$$dP = \rho d\Psi \tag{3.42}$$

which tells us that the total pressure P and density ρ are functions of Ψ alone. Furthermore, the temperature T must also be a function of Ψ as must the opacity $\kappa(\rho,T)$. From (3.42) it follows that

$$dP/dn \ \alpha \ d\Psi/dn \ \alpha \ dT/dn \tag{3.43}$$

thus,

$$F \alpha d\Psi/dn \tag{3.44}$$

With the definition of a central force potential given by

$$\Psi \alpha 1/r \tag{3.45}$$

it follows that

$$d\Psi/dn \alpha 1/r^2 \tag{3.46}$$

but this is simply the local gravity.

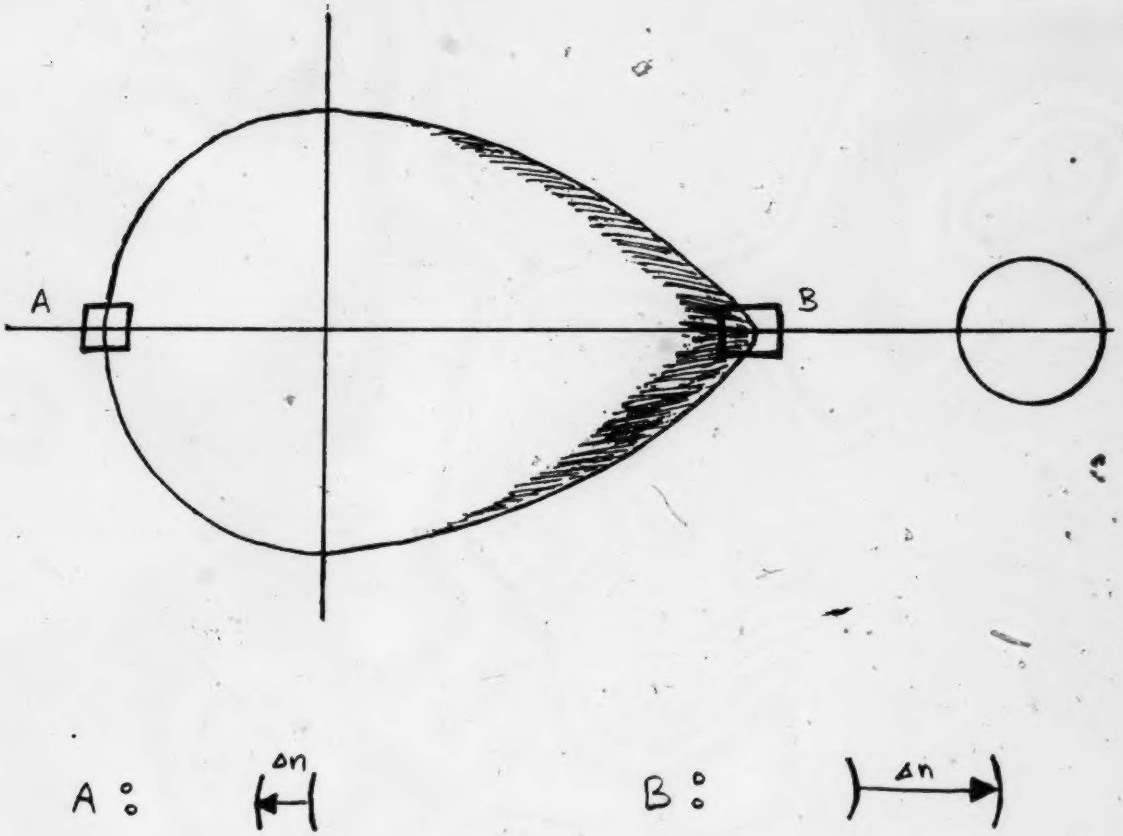
Thus, the emerging flux from B is lower than A and our star will appear cooler at B than at A (see Fig.3.6.). Recalling our general expression for emerging intensity, (3.36) we may scale H to an arbitrary point on the star (normally the pole) through the relationship

$$H = H_0\{1 + g_0((g - g_0) / g_0)\} \tag{3.47}$$

where g_0, the gravity darkening coefficient is given by

$$g_0 = h\nu/4kT(1 - \exp(-h\nu/kT)) \tag{3.48}$$

Fig.3.6. A simple illustration of gravity-darkening.
In a simplistic model of our star $d\psi / dn$
is smaller for B than for A as dn is larger.



3.4.3. Reflection Effect

Let us next consider the 'reflection effect'. Suppose that the two stars are close in proximity and they differ in their spectra (temperature). Any incident light from one star upon 'reflection' from the other star will no longer have the same energy distribution as light from the original star.This effect is not simple to model (Collins, p.438-439) for two principle reasons.

First, it is dependent on the frequency of the incident light. If one has ionizing radiation falling onto a cooler star then this is mostly absorbed by upper layers of neutral hydrogen. If the radiation is highly penetrating x-rays then it will be absorbed lower in the atmosphere and will tend to elevate the entire atmospheric temperature distribution (not just the illuminated hemisphere).

Second, it is dependent on the geometry of the reflection. Obviously the position and relative sizes of the two stars will affect the contribution of reflected light.

The treatment offered by Kopal is as follows (Kopal (1959), p.238). Consider the secondary as the illuminating component. The luminosity at an effective wavelength for this component is given by

$$(L_2)_\lambda = (J_\lambda/J_b)_2 L_2 \tag{3.49}$$

where L_2 is the total integrated light.

J_b and J_λ represent the surface brightness in the total (bolometric) and discrete light respectively.

$$\left.\begin{aligned} L_2 &= \pi a_2{}^2 J_b \\ (L_2)_\lambda &= \pi a_2{}^2 J_\lambda \end{aligned}\right\} \tag{3.50}$$

If a known fraction of light from the secondary is intercepted by the primary in the direction of the line of sight (β)

$$L = q(\beta)L_2 \tag{3.51}$$

then the amount of reflected light observable at an effective wavelength λ is given by

$$L_\lambda = (J_l/J_b)_1 L = q(\beta)(J_\lambda/J_b)_1 L_2 \tag{3.52}$$

From (2.64) and (2.67) it follows that

$$L_\lambda = q(\beta)\ (L_2)_\lambda (J_2/J_1)_b (J_1/J_2)_\lambda \tag{3.53}$$

$(J_1/J_2)_\lambda$ represents the ratio of mean surface brightness of the illuminated hemispheres of both components at the effective wavelength λ. If the stars are eclipsing this may be deduced from the ratio of the depths of the primary and secondary minima. If the stars radiate as black bodies it also follows that

$$(J_1/J_2)_\lambda = (\exp(c2/\lambda T_2) - 1)/(\exp(c2/\lambda T_1) - 1) \tag{3.54}$$

where $c2 = 1.438$ cm deg. $(J_1/J_2)_b$ is the ratio of bolometric surface brightness and for black bodies is given by

$$(J_1/J_2)_b = (T_1/T_2)^4 \tag{3.55}$$

3.4.4. Rectification

Kopal's treatment of the physical processes in a binary system is more satisfactory than that proposed by Russell. The final step in his method, rectification, still suffers from limitations in its interpretation of the relevant physics. Central to Kopal's scheme is the analysis of light variation during a cycle

$$L = L_o(1 + \delta L) \tag{3.56}$$

where L_o is an arbitrary unit of light and δL is the variation associated with the various secondary effects. Kopal goes into great detail developing an algebra (alpha-functions) to describe the dependence of the various contributing effects as a function of geometry and phase.

Nevertheless, the final prescription for the light curve is of the form

$$I_{rec} = I_{obs}(1 + C \cos 2\theta) + B \cos \theta + (1 - I_{obs})(a - b \cos \theta - c \cos 2\theta) - \ldots \tag{3.57}$$

where B and C come from the observations but a,b,c must be determined from theory. Such a determination follows from assumptions very similar to Russell's in the sense that the effects of gravity- and limb-darkening are assumed to be independent and have no effect on the out-of-eclipse variation. That is, the out-of-eclipse variation may be attributed entirely to reflection and oblateness.

3.5. Conclusion

The advent of readily available, modern computers made it possible to return to Kopal's basic principles and make direct calculations for light variation during the cycle of the star. Thus, the difficulties encountered in the rectification process may be avoided.

Notes

[1] In this very simplistic scheme the surface temperature of the star at $\theta = 0$ is related to the effective temperature by $T(\theta) = T_{eff}/2^{1/4} = 0.841 T_{eff}$ (Bowers and Deeming, p. 78)

[2] The optical opacity is related to the mean free path for a photon via $l_v = k_v^{-1}$ where $k = k_v^a + k_v^s$. k_v^a and k_v^s are the absorption and scattering opacities respectively. (Bowers and Deeming, p.70)

CHAPTER FOUR

The Wilson-Devinney Approach

4.1. Introduction

Robert Wilson and Edward Devinney published their original code in 1971 (Wilson & Devinney, 1971). MR Cygni, a W Ursa Majoris (contact) system was the subject of their first analysis. The code has been updated on several occasions to incorporate better physical models; the version used in this paper was last updated in 1985.

The WD code provides a means to investigate simultaneously the radial velocity and light variation during the cycle of an eclipsing binary system. Analysis of the radial velocity data closely follows that already outlined in Chapter Two (Method of Lehmann-Filhes). This chapter will consider the analysis of light variation during a cycle in terms of the principles detailed in chapter two.

4.2. Method of Analysis

In principle, the analysis of light variation during an eclipsing binary's cycle is straightforward. At a given phase we have

$$\text{Flux} = \sum_{\text{(contributing elements)}} \text{flux element} \tag{4.1}$$

It is only necessary to determine :

1) The geometric/physical conditions influencing the flux for a given element and,

2) The "visibility" of the element i.e. whether it contributes to the flux as viewed by an observer.

Let us consider the first of these problems; the contribution of flux for each element. In the WD code a star's surface is described as a set of discrete point sources

Each point source on a given component will then contribute a flux given by

$$\Delta F_\lambda = r^2 \sin\theta \cos\gamma * GDRI * \Delta\phi\Delta\theta / \cos\beta \qquad (4.2)$$

(Wilson & Devinney, 1971) where r is the radius to the point source from the centre of the component, θ is the longitude to the centre of the surface element, ϕ is the colatitude, γ is the angle between the local surface normal and the line of sight, β is the angle between the surface normal and the radius vector. G,D and R are the influences on the flux due to the gravity-darkening, limb-darkening and reflection effect respectively. I is the mean emergent intensity required to produce the 4π steradian luminosity of the star (see Fig.4.2).

Fig.4.1. Illustration of Grid Representation of Stars in the WD code.

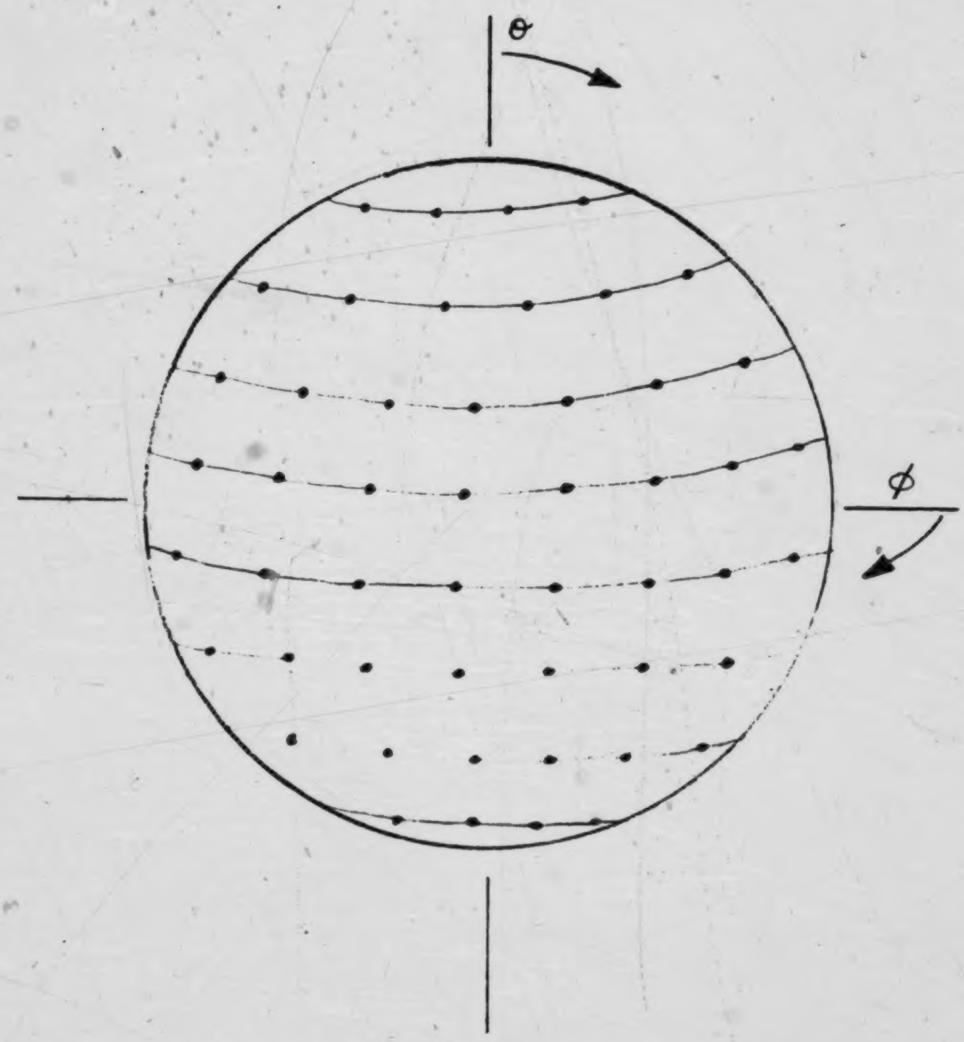
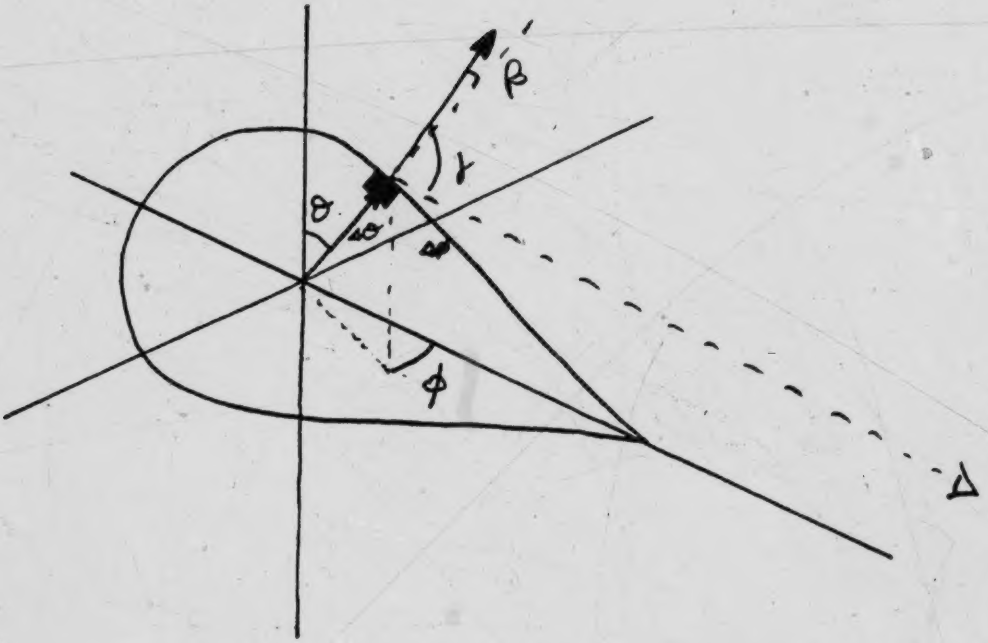


Fig.4.2. An illustration of some of the geometry of ΔF_λ.



A star which is highly distorted must have its emerging intensity scaled such that the total luminosity from the star is 4π. This is done via the I/cosβ expression and represents the total emergent luminosity normal to the star's surface after taking into account any oblateness (see Fig.4.3).

The surface brightness (SBR) is calculated once in the WD code for a given input luminosity.

It remains to calculate the influence of G,D and R on the flux. The general algorithms employed in the WD code are described in the following sections.

4.2.1. Calculation of G, Gravity-Darkening.

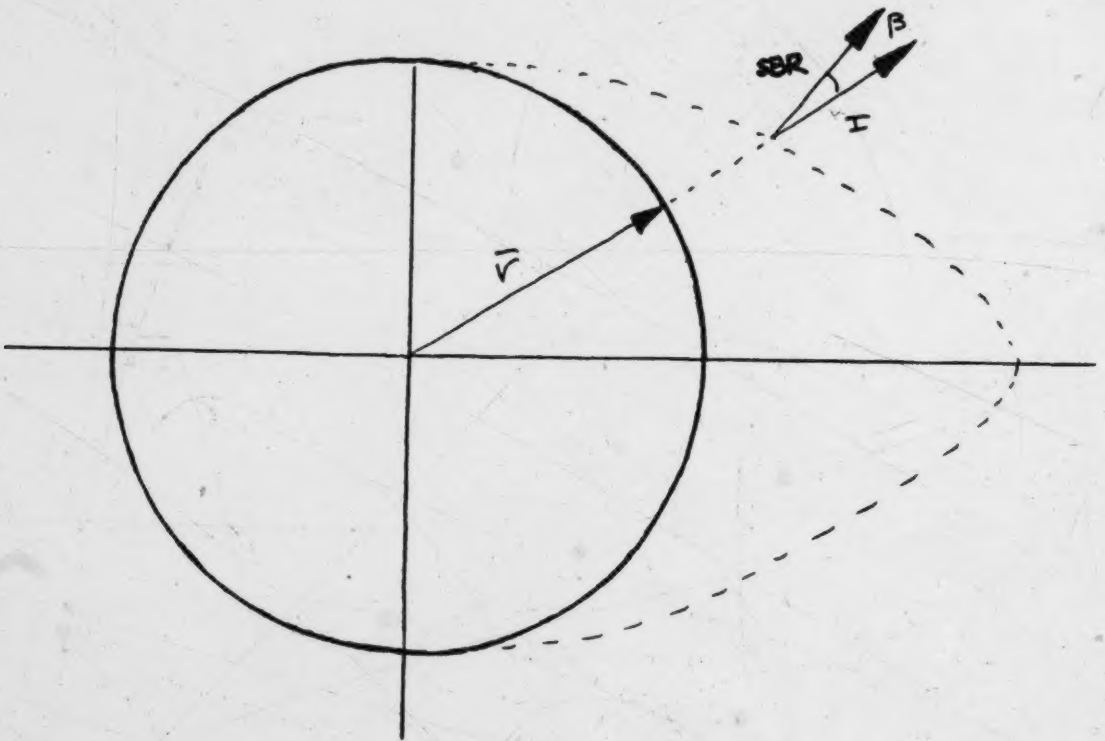
To calculate the effect of gravity darkening on our luminosity is a multiple-step process. The first parameter to consider is the ratio of bolometric flux to that at the pole

$$F_{local}/F_{pole} = (\mathrm{grad}\ \Omega)^{g} \qquad (4.3)$$

for a given surface element. g, the gravity-darkening exponent equals unity for purely radiative atmospheres as previously discussed. Later-type stars with convective atmospheres will have a g somewhat less than unity. Calculation of $(\mathrm{grad}\ \Omega)^{g}$ in the code is tedious but not difficult in the WD code. The defining equation for the Roche potentials is now

$$\Omega = 1/r + q((D^2 + r^2 - 2r\lambda D)^{-1/2} - r\lambda/D^2) + (1/2)F^2(1 + q)r^2(1 - \nu^2) \qquad (4.4)$$

Fig.4.3. The relationship between I and cos β.



(Wilson, 1979) where D is the instantaneous separation of the two components and F is the ratio of angular rotation rate to synchronous rate. To find $(\text{grad}\ \Omega)^g$ normalized at the pole we need only consider the magnitude of our potential gradient locally and at the pole. This follows as :

$$\left.\begin{aligned}(\text{grad}\ \Omega)_{\text{pole}} &= ((\partial/\partial x)^2+(\partial/\partial y)^2+(\partial/\partial z)^2)^{1/2}{}_{\text{pole}} \\ (\text{grad}\ \Omega)_{\text{local}} &= ((\partial/\partial x)^2+(\partial/\partial y)^2+(\partial/\partial z)^2)^{1/2}{}_{\text{local}}\end{aligned}\right\} \quad (4.5)$$

Details of these derivatives are provided by Wilson (Wilson, 1979) but the choice of rectangular coordinates is strictly from computational considerations. We may note that our local gravity $(\Omega\ \text{grad})_{\text{local}}$ will be greatest at the poles so that $(\Omega\ \text{grad})^g$ will always be less than or equal to unity.

As the next step in the analysis we must now calculate

$$T_{\text{local}} = T_{\text{pole}}(F_{\text{local}}/F_{\text{pole}})^{1/4} \quad (4.6)$$

which follows from Stefan's law. In the original version of the WD code, it was assumed that the eccentricity of the system was zero and thus, T_{pole}, supplied as an input parameter, would be constant at all phases. Later versions, such as the one currently in use, have an average effective temperature T_e as an input parameter. The relationship between T_e and T_{pole} is now given by

$$(T_e)_{pole} = T_e \left\{S / \int_{surface} (grad)^g dS\right\}^{1/4} \tag{4.7}$$

(Wilson, 1979).The importance of this relationship is that it allows one to vary $(T_e)_{pole}$ as a function of phase while still maintaining it as a reference point at that given phase; S is the surface area presented to the observer. Evaluation of the integral will always lead to a surface area less than or equal to S. Thus, our curly bracket in (3.7) is always greater than or equal to unity. Hence, $(T_e)_{pole}$ at a given phase is always greater than or equal to our input T_e depending on the degree of gravity-darkening (oblateness).

Finally, we must make use of Planck's law to calculate G,the ratio of local monochromatic normal emergent intensity to that at the pole. This is given by

$$(I_{local}/I_{pole})_\lambda = (\exp(c2/T_{pole})-1)/(\exp(c2/T_{local})-1)$$
$$\equiv G \tag{4.8}$$

where c2 = 1.4384 cm deg.

4.2.2. Calculation of R, Reflection Effect.

To calculate the reflection effect it was necessary for Wilson and Devinney to make certain assumptions regarding the system.

1) The irradiating component is treated as a point source.

2) The irradiated region is limited to those regions which are in view of the irradiating point source.

Within this framework the WD code calculates the reflection effect for both components. This is again a multiple-step process similar to that for gravity darkening. One ultimately wishes to find the ratio T_{new}/T_{old} where T_{new} is the local temperature corrected for reflection and T_{old} is the temperature based solely on gravity-darkening. This ratio may be found from

$$T_{new} = T_{old}((L_1/L_2)_{bolo}{}^*(\text{geometric factors}))^{1/4} \tag{4.9}$$

(Wilson & Devinney, 1971) or

$$T_{new} = T_{old}(1 + F_1/F_2)^{1/4} \tag{4.10}$$

The ratio F_1/F_2 is the ratio of irradiating flux to bolometric surface flux of the affected component. From the assumptions regarding our irradiating source and assuming our flux follows an inverse square law then

$$F_1 = L_{1bolo} \cos \alpha / r^2 \tag{4.11}$$

L_{1bolo}, the irradiating bolometric luminosity is found from the monochromatic luminosity (input parameter) after an appropriate correction, $\cos \alpha$ is the cosine of the angle between the incident radiation and the normal to the surface, r is the distance from the irradiating point source to the affected surface.

$$F_2 = (\text{grad } \Omega)^g L_{2\text{bolo}} / \int_{\text{surface}} (\text{grad } \Omega)^g \, ds \qquad (4.12)$$

F_2 is the appropriately scaled local bolometric flux at the irradiated component. From the above it follows that F_1/F_2 is given by

$$F_1/F_2 = (L_1/L_2)_{\text{bolo}} {}^* A \cos\alpha \int_{\text{surface}} (\text{grad } \Omega)^g ds / 4\pi r^2 (\text{grad } \Omega)^g \qquad (4.13)$$

(Wilson et al., 1972)

A in (4.13) is the albedo of the irradiated element. For local conservation of energy the albedo is set to unity. In general, this will hold for stars with $T_e > 7500^0$ K (Wilson, private correspondence). For cooler stars A is set to one-half.

Thus, from our new ratio of T_{new}/T_{old} one employs Planck's law to find R :

$$(I_{new}/I_{old})_\lambda = (\exp(c2/T_{old}) - 1) / (\exp(c2/T_{new}) - 1)$$
$$\equiv R \qquad (4.14)$$

4.2.3. Calculation of D, Limb-Darkening.

The WD code employs a linear law of limb-darkening given by

$$J = J_0(1 - x + x \cos\gamma) \qquad (4.15)$$

D, the ratio of "darkened" surface intensity to a normal surface intensity follows simply as

$$J/J_0 = (1 - x + x\cos\gamma) \equiv D \tag{4.16}$$

Determination of the "visibility" of an element requires a fairly complex geometric analysis. It is assumed that our coordinate system is centred at the more massive star initially i.e. at phase (θ) = 0. At each θ the star closest to the observer has its boundary represented by a Fourier series. A comparison of the projection onto the sky of this boundary with the coordinates of the individual elements of the more distant star will tell us if it is visible or hidden. A hidden element does not contribute to the observed flux. At the appropriate phase the coordinate system will be switched so as to be centred at the less massive star i.e. when it is closer to the observer.

4.3. Method of Solution

As mentioned, the WD code employs a least-squares technique to obtain a 'best-fit' curve through the photometric data. The defining equation of condition for this scheme is

$$\begin{aligned} \Delta l &= (\partial l/\partial i)\,\Delta i + (\partial l/\partial g_1)\,\Delta g_1 + \ldots \\ &= O - C \end{aligned} \tag{4.17}$$

Thus, our variation in flux between observed and theoretical (computed) values may be expressed as a linear combination of variation of relative flux due to its independent contributors.

4.3.1. Matrix Methods

In matrix form we may rewrite (4.17) as

$$Ax = b \tag{4.18}$$

where A is a matrix (m x n with $m \geq n$) representing the coefficient of variation (the $\partial l / \partial i$'s, etc.), x is a column vector representing the variation from our theoretical value, and b is a column vector representing our residual, O - C

It is only necessary to find the best x to minimize O - C. The general method of minimizing the residual follows with the observation that the least-squares solution x^* satisfies the n x n linear system

$$A^T A x^* = A^T b \tag{4.19}$$

(Johnston, p.78-79). Hence, our algorithm follows as :

i) Form $A^T A$ and $A^T b$.

ii) Solve $A^T A x = A^T b$.

This is simply solved using Gaussian elimination with pivoting (subroutine SQUARE). To obtain A one may use

$$\partial I / \partial i = (I(i + \Delta i) - I(i)) / \Delta i \text{ (asymmetric)} \qquad (4.20)$$

or

$$\partial I / \partial i = (I(i + \Delta i/2) - I(i - \Delta i/2)) / \Delta i \text{ (symmetric)} \qquad (4.21)$$

with A^T being the appropriate transpose.

For our ideal system where there are no errors due to

i) computational limitations or

ii) noisy data

we could take the solution, the vector of corrections, and simply apply them to our model parameters. However, given the limitations mentioned above it is necessary to follow an iterative procedure to obtain a final solution which minimizes the residual to within the allowed tolerances.

4.3.2. Convergence

This is a simple but tedious operation. Starting with the initial model it is easy to obtain the sum of the squares of the residuals (from input) and a set of corrections. By applying the corrections to the model and recalculating the residuals a second time it is possible to obtain a second sum of residuals squared (predicted). A good final solution will have close agreement between these two values.

To determine just how good the agreement between input and predicted solutions are it is necessary to consider the corrections in terms of their probable errors. If a parameter correction is less than or equal to its probable error then to within the uncertainty claimed there is no justification for further varying that parameter. The probable error of a parameter, the limit to within which the correct value will occur half the time is found from

$$r_1 = 0.6745\ ((pv^2)/m\text{-}u)^{1/2} \tag{4.22}$$

(Van de Kamp, p.219) where r_1 is the probable error of unit weight, p is the weight of each normal equation, v^2 is $(O\text{-}C)^2$, m is the number of normal equations and u the number of unknowns.

To determine a particular probable error it follows as

$$r_x = r_1/(p_x)^{1/2} \tag{4.23}$$

where p_x, the weight for a particular parameter is found from the appropriate diagonal of the normal equations if the off-diagonal elements are zero (subroutine DGMPRD).

Finally, one may check the determinant of the system of normal equations. This is indicative of the ill-conditioning or orthogonality of the system. A system is ill-conditioned if it has a very small determinant. This suggests there is a strong correlation between some or all of the parameters which are being varied.

4.4. Conclusion

The theoretical basis of the WD code is straightforward. The observed flux at a given phase of the binary's cycle is simply the sum of contributing surface elements after consideration of the appropriate geometry and physics. This must include the "visibilty" of the component and the effects of gravity-, limb-darkening and the reflection effect.

The computational basis of the code is a least-squares fit between observed and calculated light/radial velocity curves. This assumes that the contribution to light/radial velocity variation from the various physical and geometric parameters are independent.

CHAPTER FIVE

Using the WD Code

5.1. Introduction

In general, obtaining a solution for an eclipsing binary is a two step process. First, a synthetic light curve solution (LC) is generated from a given set of orbital and physical parameters. Comparison between this preliminary solution and observations will reveal whether the solution is reasonable or not. Second, a differential corrections program (DC) is applied to minimize the least-squares fit between observation and the starting synthetic curve.

5.2. Synthetic Light Curve (LC)

The current version of LC allows a large number of parameters to be varied; as such, it proves to be a powerful tool in the analysis of eclipsing binaries. However, it also requires careful consideration of the relevant physics to make a given problem tractable. To run LC we need an initial set of parameters. The spectral classification allows us to estimate the mass, radius and surface temperature of the component stars. It must be kept in mind that these models are based upon stars which are evolving independently. A close binary system is certainly not such a case.

Added complications due to distortions, mass transfer, etc. limit our starting model to being a rough estimate at best. Hence, the user should be prepared to vary the parameters in moderately large increments initially to become familiar with possible combinations which may fit the data. At this stage the user should be able to eliminate most of the possibilities based on known stellar structures.

Ultimately, however, the user must examine the spectroscopy and photometry in detail to determine whether a solution is reasonable. As a general rule of thumb, if any spectroscopic data differ significantly from that suggested by photometry it is usually prudent to assume the spectroscopic data are correct (an excellent example of this is detailed in the discussion on MR Cygni in the next chapter). The reason for this lies primarily in the difficulty of obtaining accurate photometric data. There are exceptions to the rule of course.

5.2.1. Varying the Initial Model Parameters

From the above considerations it is usually possible to reduce the number of variable parameters from 22 to five or seven (typically $i, \Omega_1, \Omega_2, T_1, T_2, L_1, L_2$). This parameter set includes most of the fundamental parameters, except mass, which describe the system. Second-order factors such as albedo should be fixed at this stage and should not be altered until a final solution from DC is obtained.

Simple criteria for the "goodness-of-fit" of the synthetic curve are:

i) Depths of the eclipses.

ii) Points of contact.

iii) Shape of the eclipses.

iv) Out-of-eclipse shape.

Usually it is not possible to satisfy all of the above criteria in one trial solution. There are strong correlations within the parameter set which invariably affects all portions of the curve. Variation of the inclination, i, will result in changes in the depths and widths of both eclipses. Variation of Ω_1 will primarily affect the shape and depth of the secondary eclipse and to a lesser extent it will have some influence on the primary eclipse. The converse is true for Ω_2. Variation of T_1 will affect the depth of the primary eclipse, but also the secondary eclipse depth and the out-of-eclipse variation (light level). Again, the converse is true for T_2. (See Fig.5.1.)

5.2.2. Checking the Solution

It cannot be emphasized enough that the process of using LC involves making judgements based on a visual inspection of the data and the synthetic light curve. At the end of each variation the user must generate a graph to illustrate the result. It is absolutely vital that the software used for this purpose is reliable and does not distort the shape of the LC output. Two simple rules which should be used are:

i) Keep the vertical scale approximately 50% larger than the maximum amplitude.

Fig.5.1. A simple schematic of the interdependence of some of the parameters.

$$\begin{tikzcd} & i \arrow[dl, leftrightarrow, bend right] \arrow[dr, leftrightarrow, bend left] & \\ \Omega_1 \arrow[rr, leftrightarrow, bend right] & & \Omega_2 \end{tikzcd}$$

$$T_1 \rightleftharpoons T_2$$

ii) Check that the horizontal scale is being properly divided.

Table II lists the input parameters by input cards, the effect of variation on the physics and computations.

5.3. The Differential Corrections Program (DC)

Once satisfactory agreement is reached between the synthetic model and observation it is appropriate to run a differential corrections program on LC. This will enable the user to:

i) Further refine the LC solution.

ii) Determine an estimate of the certainty of the parameters (probable errors).

Proper use of DC requires the user to recall the basic assumptions in performing the differential corrections. From

$$\Delta l = (\partial l / \partial i)\Delta i + (\partial l / \partial g_1)\Delta g_1 + \ldots \qquad (4.17)$$

we have as our basic premise that; firstly, our parameters are independent and secondly, second and higher order derivatives may be neglected (Wilson and Biermann, p.354). The first of these does not hold as we have already seen for the case of i, Ω_1, and Ω_2. To meet the second condition would require a more sophisticated code with a dramatic increase in computing time to be properly implimented.

There is also a third difficulty with DC, namely the choice of step sizes. If a step size for a parameter is too small it is possible that the solution will suffer from computational difficulties. This will be apparent as a random error. If the step size is too large then there exists the danger of overshooting the solution. This will lead to a systematic error; or a failure to converge.

The second of these problems must be neglected at this stage. Indeed, the addition of second and higher order derivatives in stellar modelling is not well understood. However, the first problem may be dealt with by the method of subsets, and the third by a prudent choice of step sizes.

5.3.1. Method of Subsets

> "... as experience shows, convergence is adversely affected not so much by a few very high correlations as by the presence of many high correlations..." (Wilson and Biermann, p.354)

The general method, as developed by Wilson and Biermann is to consider a functional set e.g.

$$l = l(i, \Omega_1, \Omega_2, T_1, T_2, L_1, L_2, \text{etc.}) \tag{5.1}$$

as composed of a number of subsets which are linearly independent.

$$l = l_1 + l_2 + l_3 + \ldots \tag{5.2}$$

where

$$l_1 = l_1(\Omega_1, T_2, L_1) \tag{5.3}$$

$$l_2 = l_2(\Omega_2, T_1, L_2) \tag{5.4}$$

$$l_3 = l_3(i) \tag{5.5}$$

To obtain a solution one steps through a set of corrections as suggested by $l_1 \Rightarrow l_2 \Rightarrow$ $\Rightarrow l_3 \Rightarrow l_1 \Rightarrow \ldots$ until all parameters within all subsets converge.

There are several precautionary notes about the formation of these sets worth mentioning. First, the correlation within a subset should be checked. After a few trials the user will become very familiar with this and know if things are working against him/her. However, a quick check of the determinant and correlation coefficients will suggest immediately if the subsets need to be further divided.

Secondly, one should only include one wavelength-dependent variable within a given subset. These consist of x_1, x_2, L_1, and L_2. If a solution to more than one light curve is being attempted simultaneously the resulting x's and L's will be the best representation for the entire range of wavelengths (Markworth, p.706).

5.3.2. Step Sizes

It is important in using a differential corrections program that a reasonable step size for correction be chosen. This avoids previously mentioned problems. The list of 19 adjustable parameters which require a step size are:

$$e, \omega, F_1, F_2, \phi_o, i, g_1, g_2, T_1, T_2, A_1, A_2, \Omega_1, W_2, q, L_1, L_2, x_1, x_2$$

In most cases a step size of 0.01 is appropriate. However, there are several important exceptions.

Firstly, the inclination, i, is usually incremented by 1 degree. For a final solution this may be reduced to 0.5 degrees and, thus, obtain a lower estimate of the probable error. Secondly, proper variation of the potentials, Ω's, and mass ratio, q, require consideration of the derivatives with respect to the radii (at periastron).$\partial\Omega / \partial r$ and $\partial q / \partial r$ are calculated as the last step in DC (subroutine ROMQSP). From the relationships:

$$\left.\begin{aligned} \Delta\Omega &= (\partial\Omega / \partial r)\,\Delta r \\ \Delta q &= (\partial q / \partial r)\,\Delta r \end{aligned}\right\} \qquad (5.6)$$

it is only necessary to decide on a size for Δr. This is usually set at 1% of the star's radius (in solar units). Thus, $\Delta\Omega$ and Δq follow immediately.

5.3.3. Input Cards

Table III contains the input parameters for DC with the appropriate annotations.

5.4. Conclusion

Any astronomer will invariably encounter some difficulties in using the WD code. Most of these may be avoided by:

i) Proper use of LC i.e. frequent examination of the graphical solutions.

ii) Reduction of the model set to the smallest number of variables broken into suitable subsets for DC.

Table II : LC Input Parameters
Card One

Parameter	Model Considerations	Computing Considertions
MODE	MODE 0-Detached binaries. -Luminosities and temperatures are not coupled	-Neither star may overfill its critical Roche lobe.
	MODE 1: - Contact binaries. - $e = 0$. -L_2 coupled to T_2.	- $g_1 = g_2$, $A_1 = A_2$, $\Omega_1 = \Omega_2$, $x_1 = x_2$, $T_2 = T_1 (a_1/a_2)^{.25g}$ Do not adjust L_2
	MODE 2: - Detached systems - L_2 is coupled to T_2.	- Best for one light curve analysis - Do not adjust L_2.
	MODE 3: - Contact systems. - $e = 0$ - L_2 is not coupled to T_2.	- Do not adjust L_2.

	MODE 4 : - Semi-detached systems. - Primary fills its Roche lobe.	-$\Omega_1 = \Omega_{1,critical}$ regardless of input value. - Do not adjust L_2
	MODE 5 : - Semi-detached systems. - Secondary fills its Roche lobe.	-$\Omega_2 = \Omega_{2,critical}$ regardless of input value. - Do not adjust L_2.
	MODE 6 : - Double-contact systems. - Both primary and secondary fill their respective Roche lobes.	-$\Omega_1 = \Omega_{1,critical}$ - $\Omega_2 = \Omega_{2,critical}$ - Do not adjust L_2.
	MODE -1 : - X-ray binaries.	- Ω_2 cannot be adjusted.
IPB	-Couples L_2 to T_2. Set IPB=0 - except for Algol-like systems.	Lack of convergence as T_2 may not be related to eclipse depth if IPB $\neq 0$.
IFAT1, IFAT2	- Models either black-body or model atmospheres for respective component.	- IFAT1,IFAT2 = 0 black-body. - IFAT1,IFAT2 = 1 model atmosphere

N1,N2	- Grid fineness (High precision parameters: e,ω,F1, F2,$_{o}$,i,$_{1}$,$_{2}$)	- The ratio of gride fineness is given approximately as the square of the ratio of radii.
N1L,N2L	- Grid fineness (Low precision).	- Slower for large N - Must check memory allocation.
PERIOD	- Orbital period.	- Units of days.
THE	- Semi-duration of x-ray primary.	- Only considered in MODE -1
VUNIT		- Units of velocities km/sec.
PHN	- Phase at which output light curve is normalized	Affects NORMLITE and MAG+K columns.
KPH	Phase increment for synthetic curves in thousandths of a period.	

IFRAD	Control parameter for output format. IFRAD=1 gives radii as function of phase. IFRAD=0 does not.

Card Two

Parameter	Model Considerations	Computing Considerations
E	- Eccentricity.	- Much faster if e = 0.
PER	- Longitude of periastron.	- Units of degrees.
F1,F2	- Ratio of angular rotation rate to synchronous rate for respective components.	
VGA	- Systemic radial velocity.	- Units of VUNIT.
PSHIFT	- Phase of primary conjunction. data to agree with observations.	- May be used to phase shift light and velocity.
XINCL	- Orbital inclination.	- Units of degrees.
GR1,GR2	- Gravity darkening for respective components. 1.0 for fully radiative atmosphere (von Zeipel). 0.25 for fully convective atmospheres (Lucy).	

Card Three

Parameter	Model Considerations	Computing Considerations
TAVH,TAVC	- Average surface temperature for the respective components.	- Units of 10000^0 K.
ALB1,ALB2	-Bolometric albedo for the respective components. (ALB1,ALB2 = 1.0 for T 7500^0 K, otherwise 0.5	
POTH,POTC	- Potentials for the respective components. - Check booklet supplied by Wilson for model lower limits.	Must check they agree with model limits.
RM	- Mass ratio, M_2/M_1	

Card Four

Parameter	Model Considerations	Computing Considerations
WL	- Wavelength of observation.	- In microns.
HLUM,CLUM	- 4π steradian luminosity for system. Estimated from relative depth of eclipses from shoulders.	- HLUM+CLUM $\approx 4\pi$
XM,XC	- Limb darkening for respective components.	
EL3	- Any third light present in system.	
ZERO	- Magnitude phase zero point, MAG+K column.	
FACTOR	- Multiplier for output flux.	

Card Five (Next NSP1/STOP/NSP2/STOP)

Parameter	Model Considerations	Computing Considerations
XLAT	- Colatitude of spot centre.	- One card for each spot. First set of spot cards before stop card are for star one - Units of degrees.
XLONG	- Longitude of spot centre.	- As above. Runs from 0^o on line-of-centres for relevant star as viewed from the top. - Units of degrees.
RADSP	- Angle subtended by spot.	- Units of degrees.
TEMSP	- Temperature of spot. -Hot spots > 1.0. Cold spots < 1.0.	- Including a spot card will generate a complete light/radial velocity curve. Normally only phases 0.0 - 0.5 are printed.

Table III : DC Input Parameters

Card One

Parameter	Model Considerations	Computing Considerations
$e, \omega, F1, F2, i, \phi_o, g_1, g_2, T_1, T_2$	- Δ's Step sizes.	- Must check step sizes. Normally 0.01 is suitable for all but i and s

Card Two

Parameter	Model Considerations	Computing Considerations
$A_1, A_2, \Omega_1, \Omega_2, q, L_1, L_2, x_1, x_2$	- Δ's Step sizes.	- Must check step sizes. Normally 0.01 is suitable for all but i and s.

Card Three

Parameter	Model Considerations	Computing Considerations
$e, \omega, F1, F2, i, g_1, g_2, T_1, T_2, A_1, A_2, \Omega_1, \Omega_2, q, L_1, L_2, x_1, x_2.$		- Control integers = 1 for variation; = 0 for no variation

Card Four

Parameter	Model Considerations	Computing Considerations
IFVC1, IFVC2		- Control integer for radial velocity curves of respective stars = 0 for no input curve; = 1 for curve.
NLC		- Control integer setting the number of light curves.
KO		- Control integer for use of scratch pad. KO = 0 : Do base set and stop. KO = 1 : Process subsets as suggested in this run Note : Must be subsets of base set stored on scratch pad. Very fast.

		KO = 2 : Calculate base set and subsets as requested.
KDISK		- Control integer for writing to scratch pad; = 0 do not write; = 1 do write.

Card Five

Parameter	Model Considerations	Computing Considerations
MODE, IPB, IFAT1, IFAT2, N1,N2	- As previously discussed for LC.	
N1L, N2L N2L,PERIOD		- Low precision grid elements. High precision grid elements are : e, ω, F1, F2, ϕ_o, i Ω_1, Ω_2
THE,VUNIT	- As previously discussed for LC.	

Card Six

Parameter	Model Considerations	Computing Considerations
E,PER,A,F1, F2,VGA,PSHIFT, XINCL,GR1,GR2.	- As previously discussed for LC.	- PER : Input in degrees but correction in radians. - A : Solar units (6.95 x 10^5 km).

Card Seven

Parameter	Model Considerations	Computing Considerations
TAVH,TAVC, ALB1,ALB2, PHSV,PCSV, RM	- As previously discussed for LC.	

Card Eight
(As needed for each radial velocity curve)

Parameter	Model Considerations	Computing Considerations
WLA	- Wavelength of observation.	- Radial velocity data.
HLA,CLA	- Relative monochromatic flux for each star.	- units of km/s
X1A,X2A	- Linear limb-darkening coefficients for each star	
SIGMA	- Estimated rms error for the observations.	

Card Nine
(As needed for each light curve)

Parameter	Model Considerations	Computing Considerations
WLA,HLA, CLA,X1A, X2A	- As described in card eight	
EL3A	- Any third light present in the system (not third luminosity).	- Correction is taken as is from output. Note : Ignore negative values.
NOISE		- Sets weight of data; = 1 inversely proportional to light level, = 2 inversely proportional to square of light level.

CHAPTER SIX

MR Cygni - A Test Case

6.1. Introduction

This first example of the use of the WD code (Wilson and Devinney, 1971) was applied to MR Cygni (MR Cyg). Motivation for choosing this system was to examine previous solutions made with the Russell model and verify the improvement in the physical model of the system. The first version of the WD code assumed several basic model constraints i.e. synchronicity, only black-body radiation, and no spots. However, its fundamental approach to treating the problem i.e. the direct calculation of gravity-darkening, limb-darkening and reflection had been incorporated.

6.2. Objectives

This chapter is a brief examination of the first use of our version of the WD code as adapted to an AT environment. Since we did not have the actual data used by Wilson and Devinney in their original study it was necessary to estimate the radial velocity and light data from the published ra~~dial~~ velcity and light curves. With this in mind, the purpose of this first trial was not to reproduce the original results exactly but to determine whether the strategy outlined in the previous chapter (LC/DC) allowed us to obtain reasonable results for our estimated data. Furthermore, it was hoped that any difficulties in using the code would be resolved at this stage.

In the LC procedure the "goodness-of-fit" was determined from a simple graphical comparison of a generated light/radial velocity curve to observed data. Variation was limited to a few key parameters (q, Ω_1, Ω_2). Criteria for judging the "goodness-of-fit" included primary eclipse shape, secondary eclipse shape and out-of-eclipse variation.

For the differential corrections procedure (DC), special attention was paid to the step sizes. MR Cyg also served to illustrate the problems with correlation between the parameters. It took several attempts to obtain good subsets which was indicated by sensible corrections as well as a solution which "appeared" to be converging.

6.3. The Initial Parameter Set

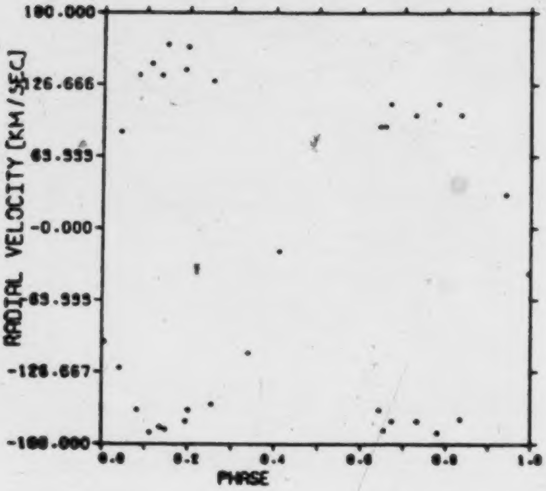
For brevity we only consider the V light curve (WL=0.5500). Fig.6.1 illustrates the data as published (Wilson and Devinney,1971) and our estimated data.

As the first step in determining the parameters Wilson considered the results due to Hall and Hardie (1969), Hill and Hutchings (1970) and Harper et al. (1935). Table IV is a synthesis of all of the relevant starting model parameters. The values for Ω_1, Ω_2, HLUM and CLUM vary slightly from those suggested by Wilson. In the case of the Ω's the tabulated values are simply the lower bound for a synchronous system with the q as suggested by Hall ($q = .56$). HLUM and CLUM are slightly different as our estimated data have a slightly different depth ratio than that used by Wilson (9.8416,2.6984).

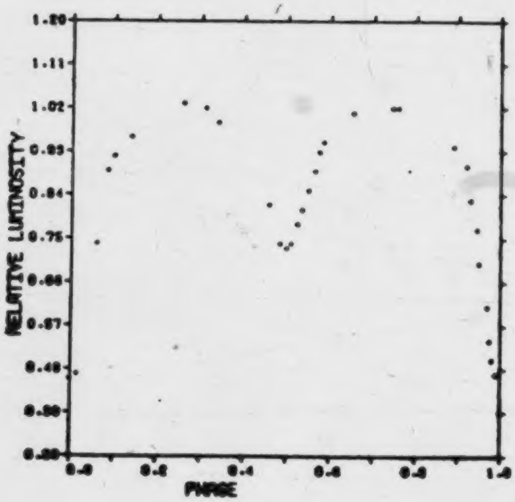
Figure withheld due to copyright restriction.

Figure withheld due to copyright restriction.

Fig. 6.1. MR Cygni (Estimated data)



Radial Velocity Data



V Photometry

One difficulty worth noting regarding MR Cyg is the estimation of the mass ratio, q. Hall claims a value of 0.56 based upon his photometric data. This does not fit in well with standard models of B-type stars (cf. Underhill and Doazan, p.56-58). A value of 0.83 had been obtained from spectroscopy (Harper et al.,1935) but the quality was very poor.

Wilson decided to retain the spectroscopic mass ratio even though the secondary is extremely faint (Wilson, private correspondence) but new spectroscopy should be performed to confirm this value. In justifying his position, Wilson notes that, firstly, the process of rectification is dubious and will not yield a true physical model of the system. Secondly, Wilson does not think the secondary has reached the main sequence but is still in the gravitational contraction stage. As such, the estimation of the luminosity ratio and hence, mass ratio claimed by Hall will not be appropriate.

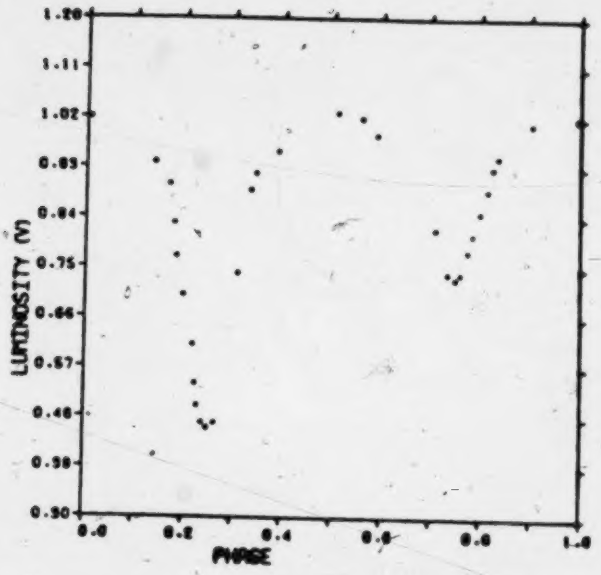
An immediate test for the code is to see which position is supported by our solution. This should be demonstrated by the simple variation of q under the assumption that the remainder of the parameters are indeed correct.

6.4. LC Results

Fig. 6.2 consists of 11 trial solutions (LC1-11)as well as the estimated data for MR Cyg. For brevity, the solutions were started near the values suggested by Wilson's final solution.

Figure withheld due to copyright restriction.

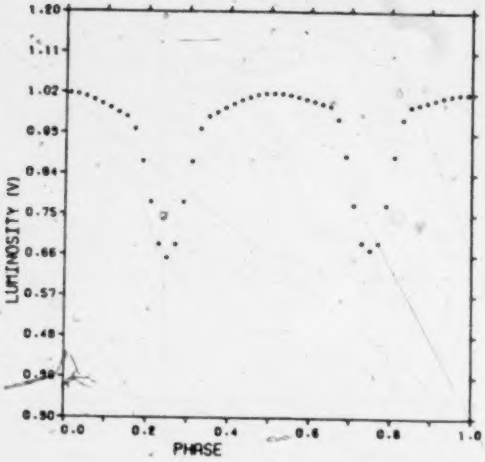
Fig. 6.2. LC Solutions (LC1-11)



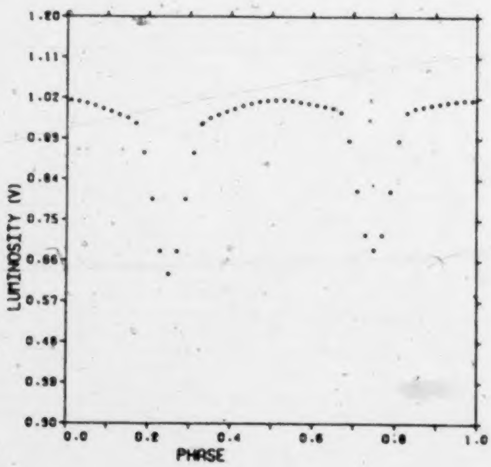
V Photometry (Estimated data)

* Solutions from LC for MR Cyg were judged by three criteria. Agreement with primary eclipse depth and shape, agreement with secondary eclipse depth and shape, and out-of-eclipse variation. Judgement of shape included both amplitude and rate of variation of the shape.

Fig. 6.2. LC Solutions (LC1-11)

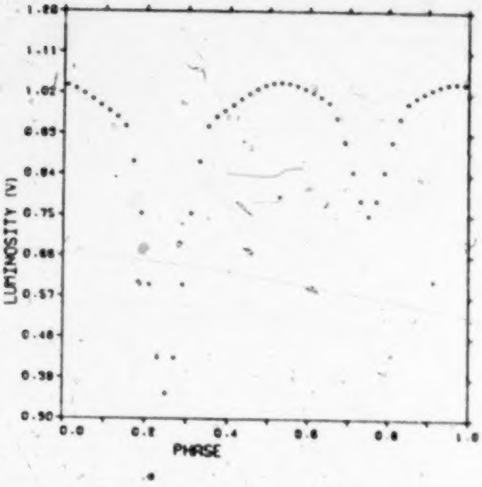


Sol.	Parameters	Primary	Secondary	Out-of-Eclipse
1	$\Omega_1=3.5$ $\Omega_2=3.0$ $q=0.56$	Underestimates Good fit	Overestimates Good fit	Underestimates Poor fit

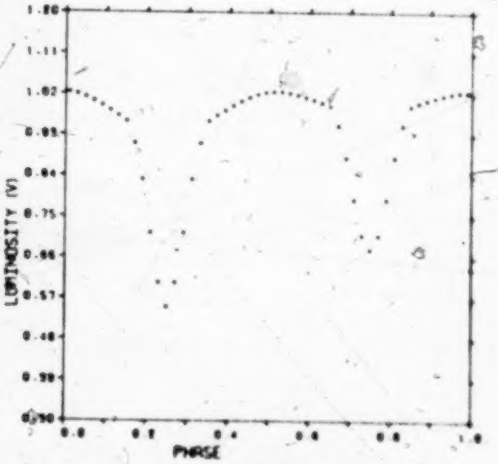


Sol.	Parameters	Primary	Secondary	Out-of-Eclipse
2	$\Omega_1=3.5$ $\Omega_2=3.2$ $q= .56$	Underestimates Gentle slope	Overestimates Steep slope	Underestimates Fair fit

Fig. 6.2. LC Solutions (LC1-11)

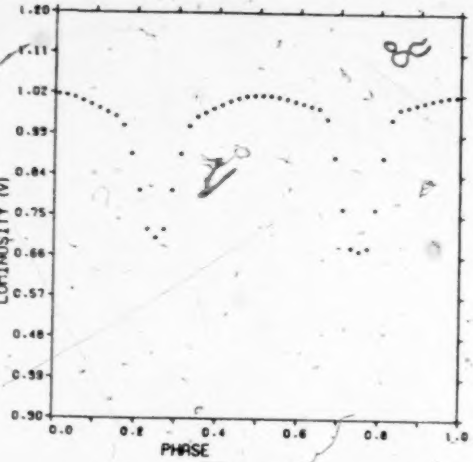


Sol.	Para-meters	Primary	Secondary	Out-of-Eclipse
3	$\Omega_1=3.5$ $\Omega_2=3.4$ $q=.56$	Overestimates Steep slope	Underestimates Good fit	Underestimates Good fit

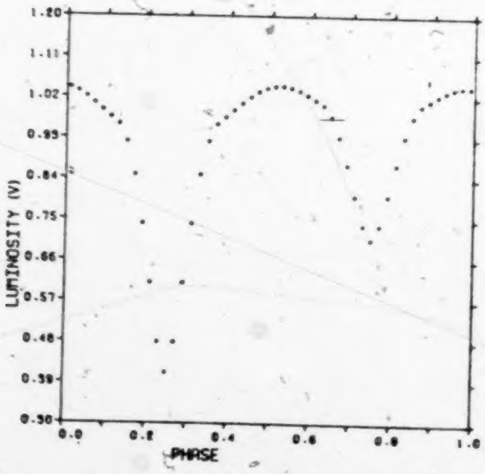


Sol.	Para-meters	Primary	Secondary	Out-of-Eclipse
4	$\Omega_1=3.5$ $\Omega_2=3.8$ $q=.56$	Underestimates Poor fit	Underestimates Good fit	Underestimates Good fit

Fig. 6.2. LC Solutions (LC1-11)

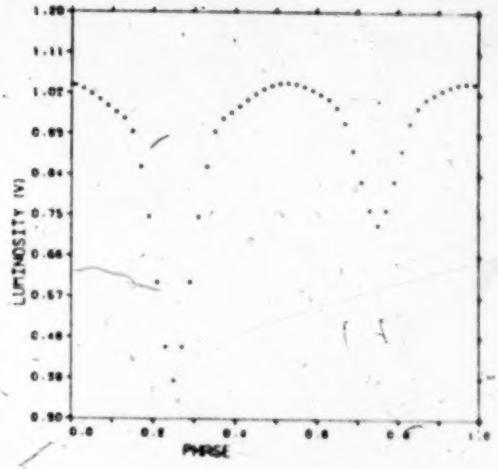


Sol.	Para-meters	Primary	Secondary	Out-of-Eclipse
5	$\Omega_1=3.5$ $\Omega_2=4.0$ $q=.56$	Underestimates Shallow	Overestimates drastically Poor fit	Very poor Shoulders much too high

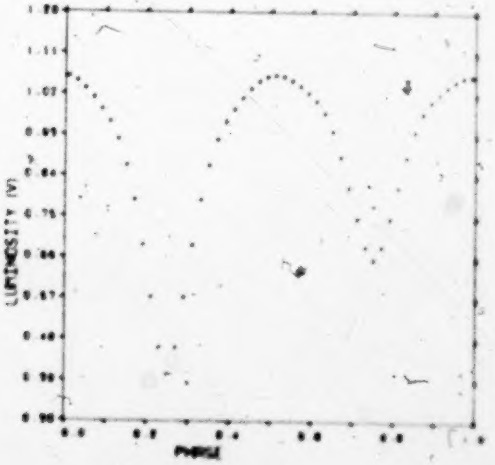


Sol.	Para-meters	Primary	Secondary	Out-of-Eclipse
6	$\Omega_1=3.7$ $\Omega_2=3.0$ $q=.56$	Overestimates	Excellent fit	Fair fit

Fig. 6.2. LC Solutions (LC1-11)

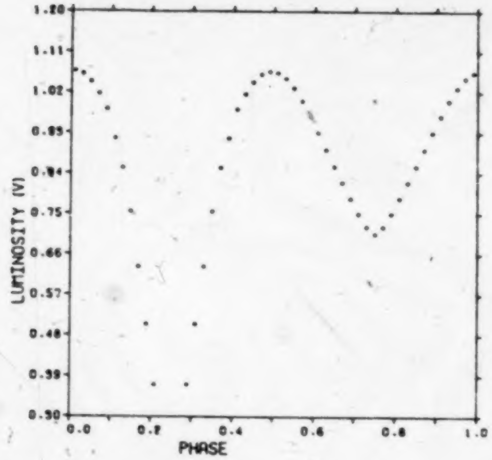


Sol.	Parameters	Primary	Secondary	Out-of-Eclipse
7	$\Omega_1=3.9$ $\Omega_2=3.0$ $q=.56$	Overestimates slightly	Excellent fit	Fair fit

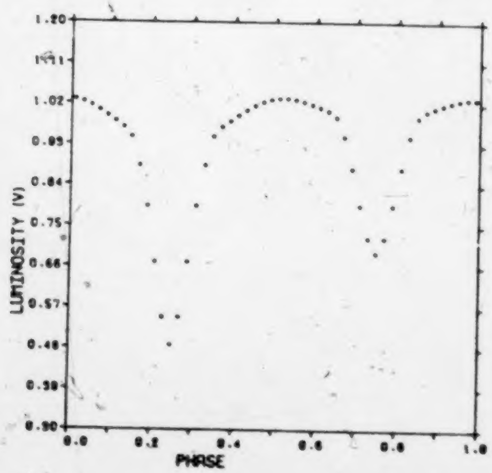


Sol.	Parameters	Primary	Secondary	Out-of-Eclipse
8	$\Omega_1=3.8$ $\Omega_2=3.9$ $q=.56$	Overestimates Shoulders much too high	Overestimates Poor fit	Poor fit

Fig. 6.2. LC Solutions (LC1-11)

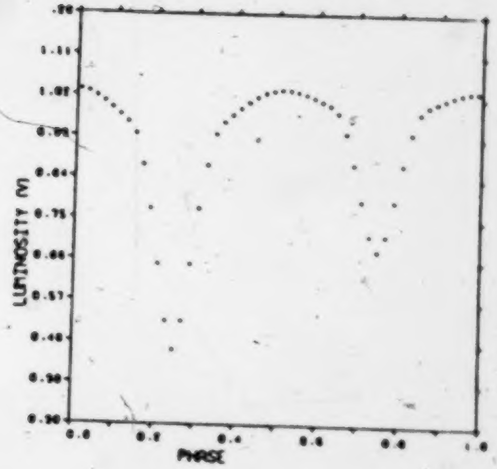


Sol.	Parameters	Primary	Secondary	Out-of-Eclipse
9	$\Omega_1=3.5$ $\Omega_2=3.0$ $q=.83$	Overestimates drastically Too deep	Poor fit	Poor fit



Sol.	Parameters	Primary	Secondary	Out-of-Eclipse
10	$\Omega_1=3.5$ $\Omega_2=3.0$ $q=.83$	Underestimates	Underestimates Good shape	Good fit

Fig. 6.2. LC Solutions (LC1-11)



Sol.	Parameters	Primary	Secondary	Out-of-Eclipse
11	$\Omega_1=3.8$ $\Omega_2=3.9$ $q=.83$	Excellent fit Shoulders right height	Good fit Shoulders right height	Excellent fit Right shape at all phases

Preliminary examination of the solutions closest to that suggested by Wilson i.e. $\Omega_1 = 3.8, \Omega_2 = 3.9$ strongly suggest that the spectroscopic q is indeed correct. The interdependence of the various parameters is also clearly illustrated. For example, comparing the effect of the variation of Ω_2 on the primary eclipse at different q values demonstrates the interrelationship between q and Ω_2.

6.5. DC Results

Starting with the best LC solution it was assumed that a properly working DC would make the appropriate corrections such that the DC solution approached the final solution as suggested by Wilson.

There were two phases to this analysis. Solutions DC(1-9) were based upon the following strategy :

i) Vary the entire base set ($i, \Omega_1, \Omega_2, L_1, L_2, x_1, x_2$).

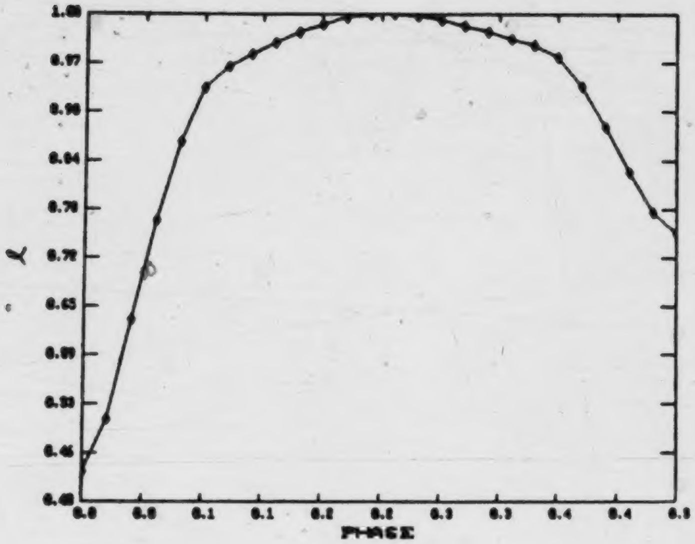
ii) Examine all combinations of subsets and determine the best possible correction set based upon agreement between the predicted and input residuals.

At this stage there is no way to judge how good the agreement is. A large subset will probably have a larger residual than a smaller set by virtue of having more contributing elements. It does not automatically follow ,however, that the larger subset is poorer in its estimation of corrections than the smaller subset. Furthermore, the step size will have some influence on the size of the residuals. Thus, there is a certain amount of guess work in the initial use of DC.

For the first attempts at a solution for DC (DC1-9) the criteria were selection of the largest subsets with the smallest percentage deviation between the predicted and input residuals. This is not the proper approach. As it turned out most of the suggested corrections were for the subset (Ω_1,Ω_2,L_1). However, the correlation is strong within this set. The reason for the close agreement between residuals lay with inappropriate step sizes. For brevity only (DC1-4) are illustrated.

The second set of solutions (DC13-16) was made after a readjustment of the step sizes. The approach towards subsets was also slightly changed. For each iteration DC only calculated a designated subset rather than the entire base set and many subsets. This saved on computing time dramatically. However, the selection of a proper subset was still not well known at this time with the result that some subsets were strongly correlated.

Fig. 6.3. DC Solutions (DC1-4)



DC1 (Ω_1, Ω_2, L_1)

Parameter Corrections

Ω_1 = 2.5584 > p.e.

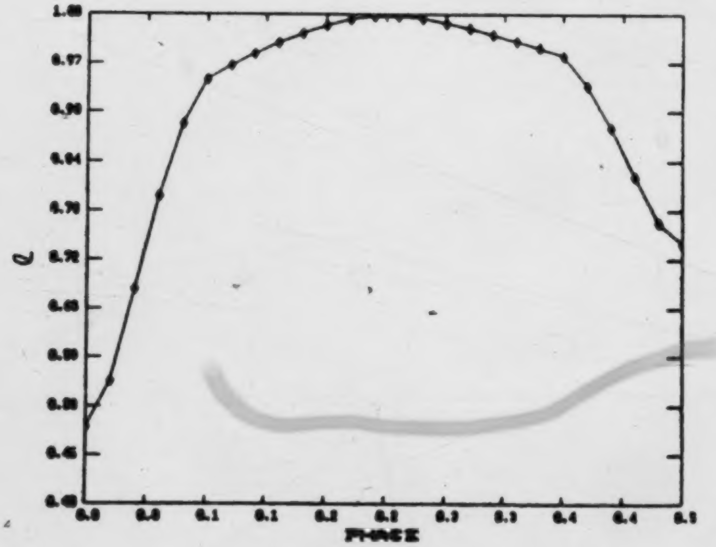
Ω_2 = .1518 < p.e.

L_1 = .0

Residuals

Input = 40.0194

Predicted = 32.963



DC2 (Ω_2, L_2)

Parameter Corrections

Ω_2 = .2749 ≈ p.e.

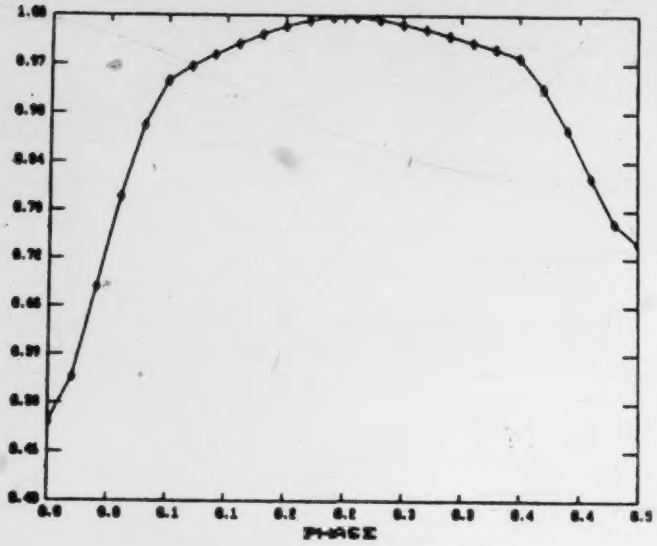
L_1 = .0

Residuals

Input = 40.0194

Predicted = 39.8615

Fig. 6.3. DC Solutions (DC1-4)



DC3 (Ω_1, Ω_2, L_1)

Parameter Corrections

Ω_1 = - .0083 <<< p.e.

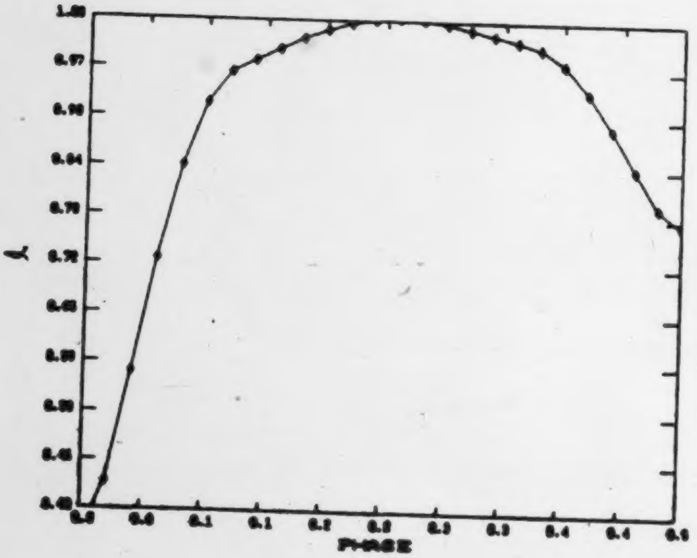
Ω_2 = .0

L_1 = - .0267 ≈ p.e.

Residuals

Input = .6291

Predicted = 6288



DC4 (Ω_1, Ω_2, L_2)

Parameter Corrections

Ω_1 = .1041 ≈ p.e.

Ω_2 = -.3407 < p.e.

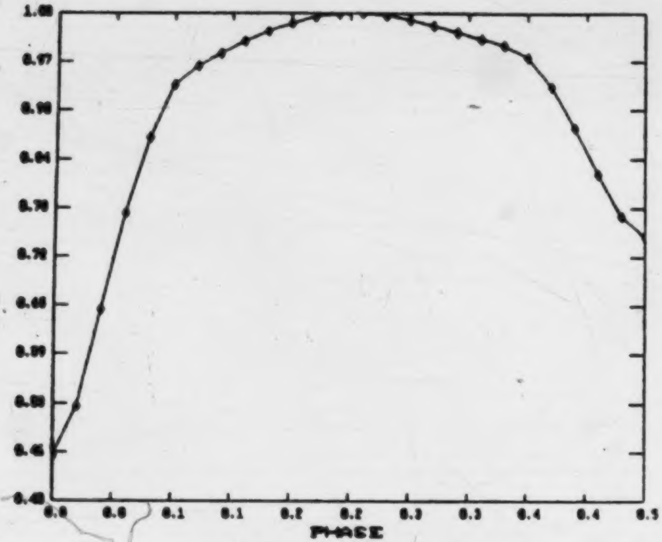
L_2 = .0023 < p.e.

Residuals

Input = .2832

Predicted = .1242

Fig. 6.3. DC Solutions (DC13-16)



DC13 (i, L_1, L_2)

Parameter Corrections

i = -.2717 > p.e.

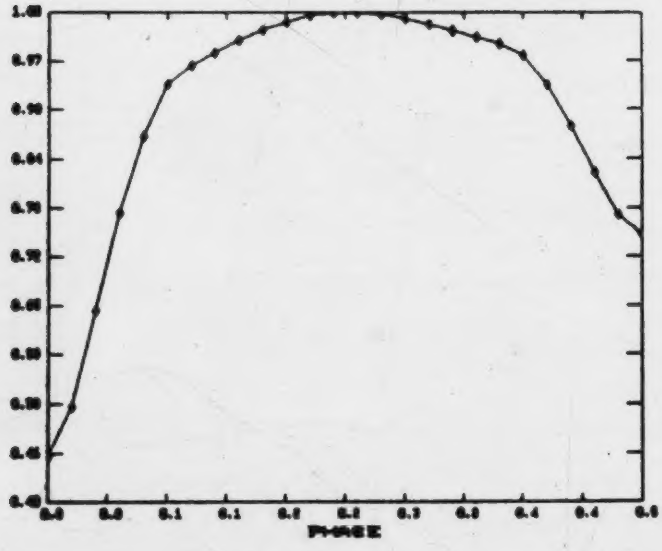
L_1 = -.0003 << p.e.

L_2 = -.0018 p.e.

Residuals

Input = .0031

Predicted = .0025



DC14 (i, Ω_2, L_2, X_2)

Parameter Corrections

i = .0

Ω_2 = .0135 > p.e.

L_2 = -.0092 << p.e.

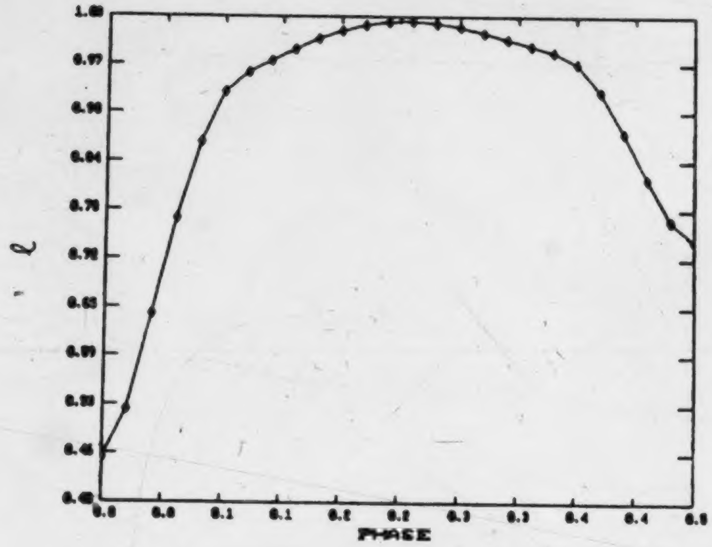
X_2 = -.0242 ≈ p.e.

Residuals

Input = .0019

Predicted = .0017

Fig. 6.3. DC Solutions (DC13-16)



Parameter Corrections

$i = .0$

$\Omega_2 = .0$

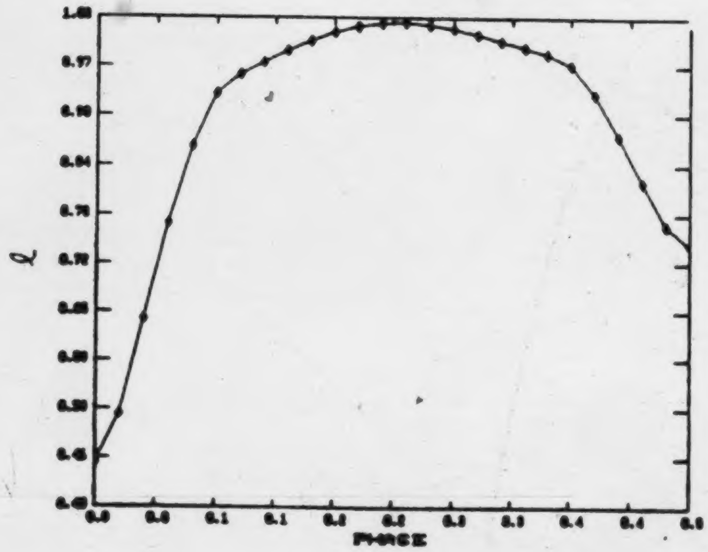
$L_1 = -.2125 \approx$ p.e.

Residuals

Input = .4061

Predicted = .3528

DC15 (i, Ω_2, L_1)



Parameter Corrections

$i = .1317 \approx$ p.e.

$L_1 = -.0278 \approx$ p.e.

Residuals

Input = .0039

Predicted = .0039

DC16 (i, X_1)

Nevertheless, the values for input and predicted residuals were significantly reduced (proper step size) and the program appeared to be approaching Wilson's solution. At this point there occurred a problem which was not detected and reoccurred in the first trials for HD219634. Selection of the control parameter IPB=1 decouples L_2 from T_2. Normally, L_2 and T_2 would be related via a black-body approximation or from a stellar atmosphere model. If you remove this constraint then it is necessary to assign the values of L_2 and T_2. Failure to do this results in an underdetermined system. Unfortunately, due to time considerations it was not possible to let DC complete its analysis of MR Cyg when this problem would have become apparent.

6.6. Conclusion

Analysis of MR Cyg proved invaluable in learning how to use the WD code. The utility of using LC is clearly demonstrated as it allows the user to quickly check the reasonableness of a solution and examine the correlation between the parameters. Furthermore, proper selection of subsets and step sizes was clearly demonstrated for the proper use of DC.

CHAPTER SEVEN

Analysis of HD219634

7.1. Introduction

Analysis of HD219634 followed standard observational procedures. First, there was the collection of photometric data. Secondly, obtaining a solution for the light curve (V and B photometry), and thirdly a comparison of the results with current stellar models.

7.2. Photometric Data

Photometric data for HD219634 were collected at the Devon Astrophysical Observatory during the period September 1987 to December 1988. The technique employed was a two channel differential method. The program star, HD219634, was centred in Channel A. The second channel, Channel B, was used to simultaneously record the signal from a non-variable standard star, SAO20532. These observations were grouped into sets of UBV filters (centred at approximately 3500λ,4500λ, and 5500λ respectively)with each filter receiving 3 integrations of twenty seconds duration. A sky count for each filter was obtained at the beginning of each set. Periodically, throughout the night, the comparison star was compared to a third star (the check star), SAO20526, to test its constancy over the observing interval.

A differential measure of HD219634 relative to the comparison star may then be found from

$$m_2 - m_1 = 2.5\log_{10}(b_1/b_2) \tag{7.1}$$

which represents the difference in magnitude between HD219634, m_2, and our comparison star, m_1. b_1 and b_2 are the actual counts recorded by the photomultiplier tubes for the respective stars. It is necessary prior to this calculation to remove the background sky contribution in each channel. This was done using a standard reduction program (HISP).

Since the data were gathered over many nights it was necessary to phase them. A program, PHASER (see Appendix B for details), was written which takes the data portion of the HISP output and phases the data from a reference Julian date, T_0 = 2443334.945, the spectroscopically determined epoch of mean longitude, with a period of P = 2.391253 days. This program also binned the data in groups of three to reduce some of the noise within a given observing set.

The two channels used in the observations were not identical. Channel A was an EMI 9840 tube while Channel B was an EMI 9826 tube. The response curves for the two channels are not identical and the relative sensitivities are a function of the filter wavelength. Thus, a final correction to account for the disparity in the response of the tubes is necessary.

This correction, the gain correction, assumes that all observations made with Channel B (the check star) are correct. One must scale the results gathered in Channel A to correct them as if they were being gathered in Channel B. To this end, periodically throughout the night the comparison star was observed through Channel A. The difference in the values for the check star as observed in the two channels would be a measure of the necessary scaling for Channel A. This is given by

$$\Delta m_{gain} = 2.5\log_{10}(b_{Channel\ A}/b_{Channel\ B}) \qquad (7.2)$$

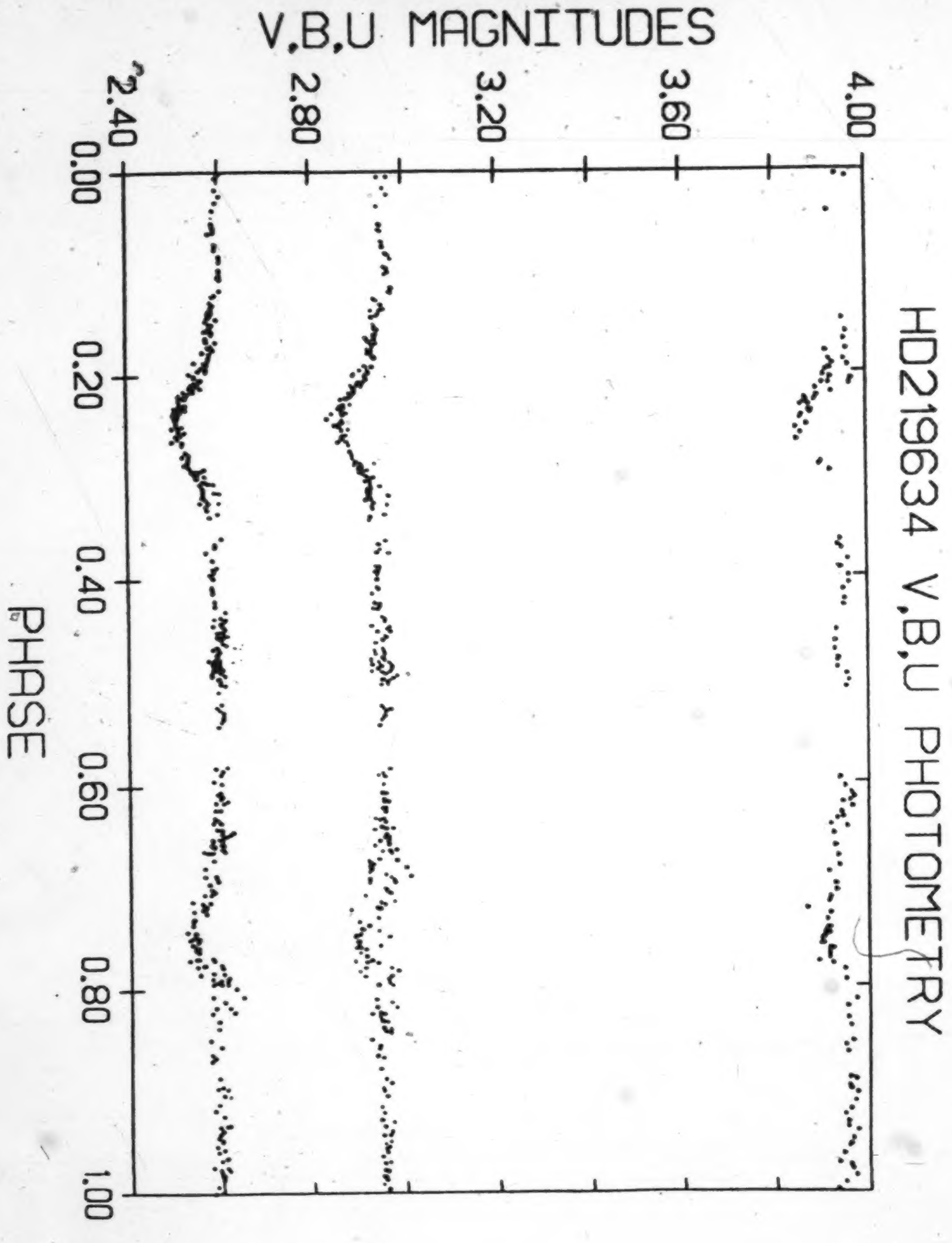
A program, CSPLINE.GAIN (Appendix B), was written to perform this correction.

Fig. 7.1 displays the "raw" data gathered for HD219634. These data have been fully reduced, i.e. sky and gain corrections have been applied. Due to computing considerations every fifth point is illustrated.

7.3. Sources of Error

In any observational program of long duration there are sure to be problems of inconsistency in results due to experimental errors. Whether these errors are due to systematic or random sources it is necessary to develop a standard observing sequence to try and minimize them.

Fig.7.1. Photometry for HD219634.



7.3.1. Systematic Sources of Error

At times these errors are very hard to detect and quantify. Some of the more obvious potential sources of error include:

i) Response function of the two tubes. Unfortunately, there will be a difference in the responses of the two tubes which will be a function of wavelength. It is believed that good gain corrections will minimize this.

ii) Colour differences of the two stars. The comparison star, SA020532, is a much cooler star than HD219634 and as such emits most of its energy at longer wavelengths. During an observation, as the stars traverse the sky there will be a loss of light as the airmass increases. This loss is not linear but is a function of the incident wavelength. This second-order colour correction is assumed to be negligible as the difference in airmass between the two stars at all times is small.

iii) Imperfections in the observing equipment. These would include such diverse problems as a poorly figured mirror (as well as dirt and scratches on the mirror surface), variation in the light path between the two channels and poorly culminated images. These errors are difficult to assess but it is assumed that these contributions would be small since the optical paths for the two channels diverge by less than one-half degree. NOTE: During the observing sequence (Sept. '87 - Dec. '89) a small light block was placed over the mirror to improve the images (the result of the poorly figured surface). This resulted in better images but with reduced counts in both channels. This reduction should not affect the differential photometry.

iv) Offset of comparison star. At the beginning of each night channel B was offset and centred on the comparison star. This offset would change slightly from night-to-night. Careful centring of channel B should minimize this problem.

7.3.2. Random Sources of Error

Random errors may have been introduced by one, or more, of the following:

i) Poor centring. Each observation required the observer to centre the object of interest in the aperture. At times, depending on the position of the telescope, the tracking of the telescope, and the alertness of the observer, this could be a significant source of error. Usually, it was very apparent if the centring was poor as the counts for the star, would be anomalous, that is, significantly smaller than normal, or changing in a rapid and systematic way.

ii) Poor sky conditions. This was by far the most difficult source of error to detect and estimate. Contributing factors to this source of error include variable cloud cover, haze, moonlight, aurora and light pollution from nearby population centres. The great advantage of the differential method over a classical one channel method is the ability to work under poorer sky conditions. Normally, it was possible to work through a partly cloudy sky provided there was no source of third light. That is to say, the 2-channel system compensates for a grey absorber. This greatly increased the number of possible observing nights. Any nights that were in doubt were hastily discarded.

A second difficulty with poor sky conditions is the effect it has on the gain corrections. It is necessary to have good, i.e. "photometric", sky conditions to obtain true corrections. On some nights it was possible only to collect two or three gains for the session rather than a set covering the entire night. For those nights when it was judged these gains were reliable it was possible to extrapolate and/or interpolate a set to cover the whole night. This required a minimum of two gains separated by a substantial amount of time.

iii) Stability of Channel B. It was noted during the observing period that Channel B could be noisy at times. Smoothing of the data within a set should reduce this to a satisfactory level. It would be best in the future to replace this tube but for the current work it was judged to be acceptable.

7.3.3. Estimation of Total Error

By being critical with the data sets i.e. discarding those nights with extremely poor sky conditions, poor gain coverage, centring, etc. it was possible to reduce the data to a reliable but still substantial set. (see Fig. 7.1) It was judged that the minimum uncertainty within this data set would be of the order of 0.5% in total from all contributing elements. This is certainly an optimistic estimate. It was hoped that by prudent selection of data that any real photometric variation of this magnitude would be detected.

7.4. Preparation for the WD Code

Preparation of the data for HD219634 for solution by the WD code required that the input magnitudes be converted to input fluxes. This is given by

$$\Delta I = 10^{(.4\,\Delta m)} + \text{offset} \tag{7.3}$$

where the offset is an arbitrary value which may be used to normalize the light curve at a given phase. For convenience, phase zero was chosen as it appeared to be relatively quiet and had good coverage.

The second consideration for preparation of data was computing time. With some 1500 data points in the V-filter alone it was necessary to bin the data into normal points to reduce computing time. A program, AVE.DAT (see Appendix B) was written which varied the bin size and calculated the number of entries per bin. The bin size was typically 0.01 of a phase with finer gradation (.005) during the eclipses.

7.4.1. The Initial Model Parameters

To construct the initial model for HD219634 we have two sources of information. First the spectroscopic work (Gulliver et al., 1985) provides invaluable information on the lower bounds of the physical/geometric makeup of the system. Secondly, from current stellar theory (cf. Underhill and Doazan, 1982) we assume starting masses, radii, etc. Table V lists the results from Gulliver et. al. Arguments presented in that paper suggest their spectroscopic "solution 3" is best representative of the system.

The HD classification of HD219634 is B8 but Gulliver et. al. suggest the primary is a BO dwarf based upon their spectroscopic data. Assuming this is correct, the mass for a BOV star (M_1) is $16.8 \pm 4.3 M_{\odot}$ (Underhill & Doazan, p. 36). Using the spectroscopic mass ration, $q = .351$ we have $M_2 = 5.9 M_{\odot}$ which suggests it is a B4-

Table V Spectroscopic Data for HD219634 (Gulliver et. al., p.257).

Orbital Element	Solution-1	Solution-2	Solution-3
γ(km s^{-1})	-26.8 ±1.1	-26.5 ±4.3	-25.7 ±0.9
K_1 (km s^{-1})	86.4 ±1.6	88.1 ±3.3	87.0 ±1.2
K_2 (km s^{-1})	——	249.0 ±3.1	247.5 ±3.0
T_o (JD2440000+)	3334.959 ±0.008	3334.924 ±0.027	3334.951 ±0.006
e	0.016 ±0.020	0.018 ±0.037	0.013 ±0.015
ω(°)	180.1 ±62.3	299.1 ±130.8	222.5 ±66.7
a_1 sini (10^6 km)	2.84 ±0.05	2.90 ±0.11	2.86 ±0.04
a_2 sini (10^6 km)	——	8.19 ±0.10	8.14 ±0.10
M_1 $\sin^3 i$ (Θ)	——	7.02 ±0.25	6.87 ±0.21
M_2 $\sin^3 i$ (Θ)	——	2.48 ±0.14	2.41 ±0.07
Mass Function (Θ)	0.160 ±0.009	——	——
Mass Ratio (M_1/M_2)	——	2.83 ±0.11	2.85 ±0.05

B7 star. The temperatures for these stars would be $30000^{o}K$ (T_1), and 17000-$12500^{o}K$ (T_2) (ibid, p. 55-56). Returning to solution 3 we have from $M_1 \sin^3 i(O) = 6.87$ with $M_1 = 16.8\ M_O$ that $i = 47.9^{o}$. From $a_1\sin i(10^6 km) = 2.86$ and $a_2\sin i(10^6 km) = 8.14$ we have $a = (a_1+a_2) = 21.3\ R_O$.

L_1 and L_2, the relative luminosities are estimated from the depths of the shoulders with the constraint that $L_1 + L_2\ 4$. Thus, $L_1 = 7.733$ and $L_2 = 4.833$. XH and XC are found in Kopal's table (Table 1) to be XH = .65, XC = .60.

As both are relatively hot it should be expected that radiation will be the most efficient way to transport energy from the star (fully radiative atmosphere). Thus, our gravity darkening coefficients, $g_1 = g_2 = 1$. Furthermore, the hotter atmospheres will scatter without absorption (Compton scattering) which suggests our albedos, ALB1 = ALB2 = 1. from conservation of energy. The critical potentials for the components stars assuming synchronization are $\Omega_1 = 2.5743$ and $W_2 = 2.3584$ (Wilson's booklet). Table VI summarizes the starting parameters for HD219634.

Fig. 7.2 illustrates the starting solution for the V photometry as generated by LC. Comparing this to Fig. 7.1 we can see that the model curve and observations appear quite dissimilar. Unfortunately, when plotting the data using the default set up for the graphing package EASYAS distorts the horizontal scale (as illustrated in Fig. 7.2). The result was the curve generated by LC was much closer to the data in appearance than it actually was.

Table VI Starting Parameters for HD219634

(Trial One)

Geometric Parameters

$a = 21.3R_\odot$

$i = 47.9^o$ (varied)

$P = 2.3913^d$

$\omega = 222.5^o$

Physical Parameters

$q = .3509$ (fixed)

$\Omega_1 = 2.6000$ (varied)

$\Omega_2 = 2.9000$ (varied)

$L_1 = 7.7300$ (varied)

$L_2 = 4.8364$ (varied)

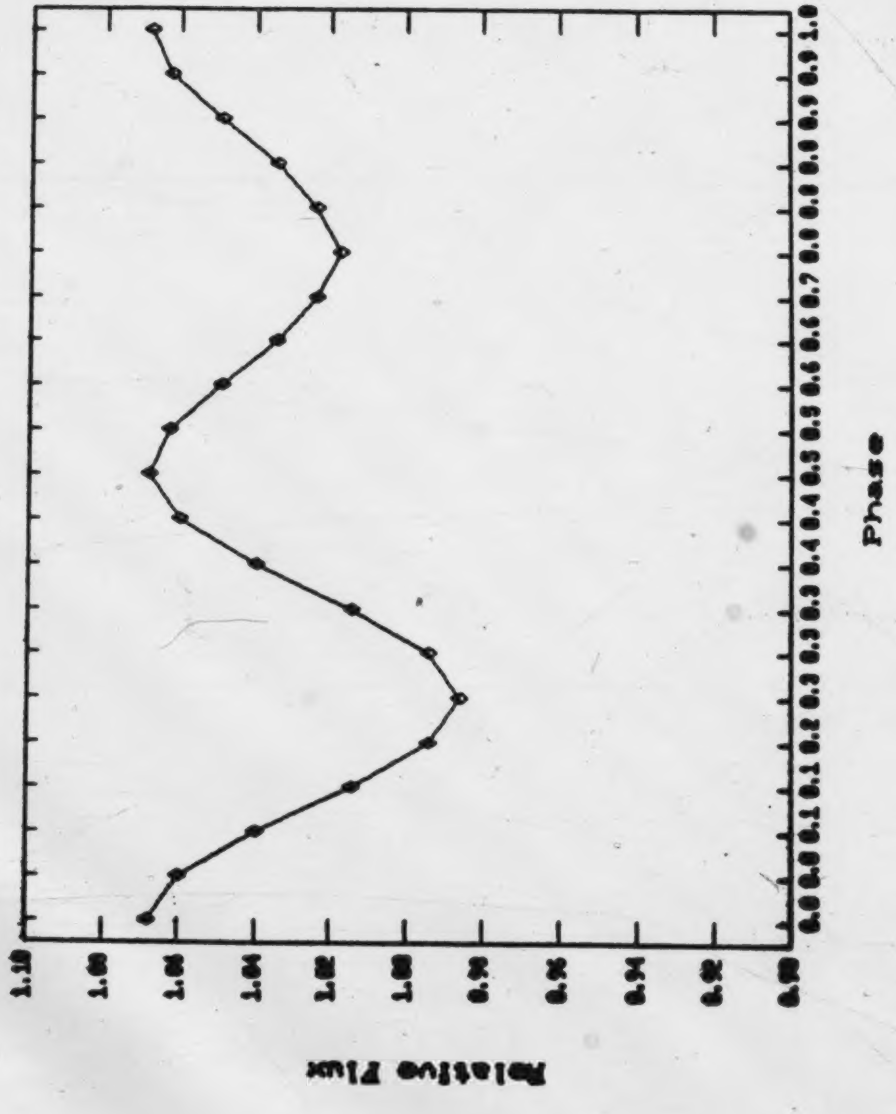
$x_1 = 0.6500$ (fixed)

$x_2 = 0.6500$ (fixed)

$T_1 = 3.0000$ (fixed)

$T_2 = 1.6000$ (varied)

Fig.7.2. Starting Solution for HD219634
as suggested by LC.



7.4.2. Trial One

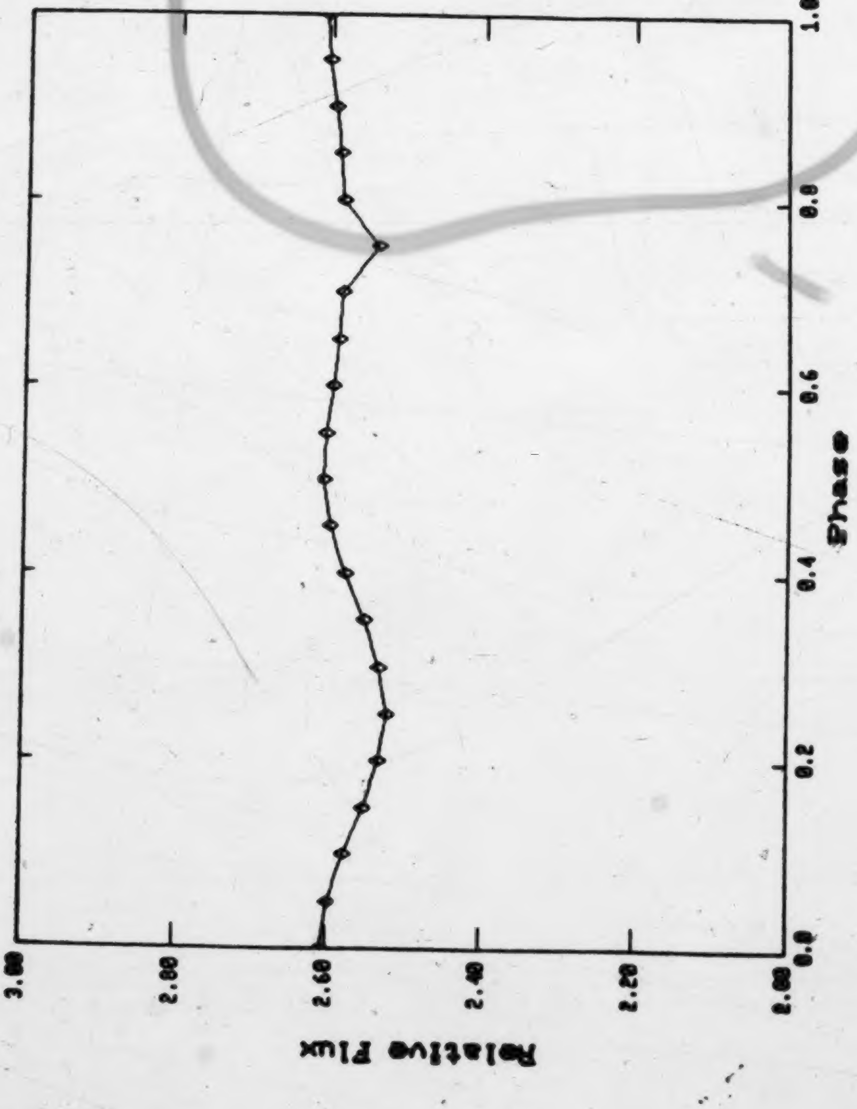
Nevertheless, starting with the above solution from LC a set of corrections was obtained from DC. The parameter subsets were:

$$\begin{array}{ll} \text{A)} & \left.\begin{array}{l} i \\ T_1 \\ \Omega_2 \end{array}\right\} \\ \text{B)} & \left.\begin{array}{l} \Omega_1 \\ L_2 \end{array}\right\} \\ \text{C)} & L_1 \\ \text{D)} & T_2 \end{array}$$

with all other parameters being held constant at their theoretical values.

During the first four calibrations the corrections seemed reasonable. The most dramatic was that for inclination which demanded an increase of 10^{o} on the first iteration. This made sense as the starting inclination of 48^{o} is low. After four complete iterations the graph scale was calibrated to give proper illustrations of the results. Fig. 7.3 illustrates the solution and data at this stage.

Fig. 7.3. End of Fourth Iteration (Trial One).



We can see that the secondary fit is good but the primary fit is exceedingly poor, as is the out-of-eclipse variation. At this point the DC code had to make a decision in performing its fitting. Either it could smooth the secondary eclipse or sharpen the primary eclipse. The first choice, the apparently incorrect choice, was taken. This suggests that the starting solution must be reasonably close to the correct solution to avoid finding and claiming local minima as the final solution.

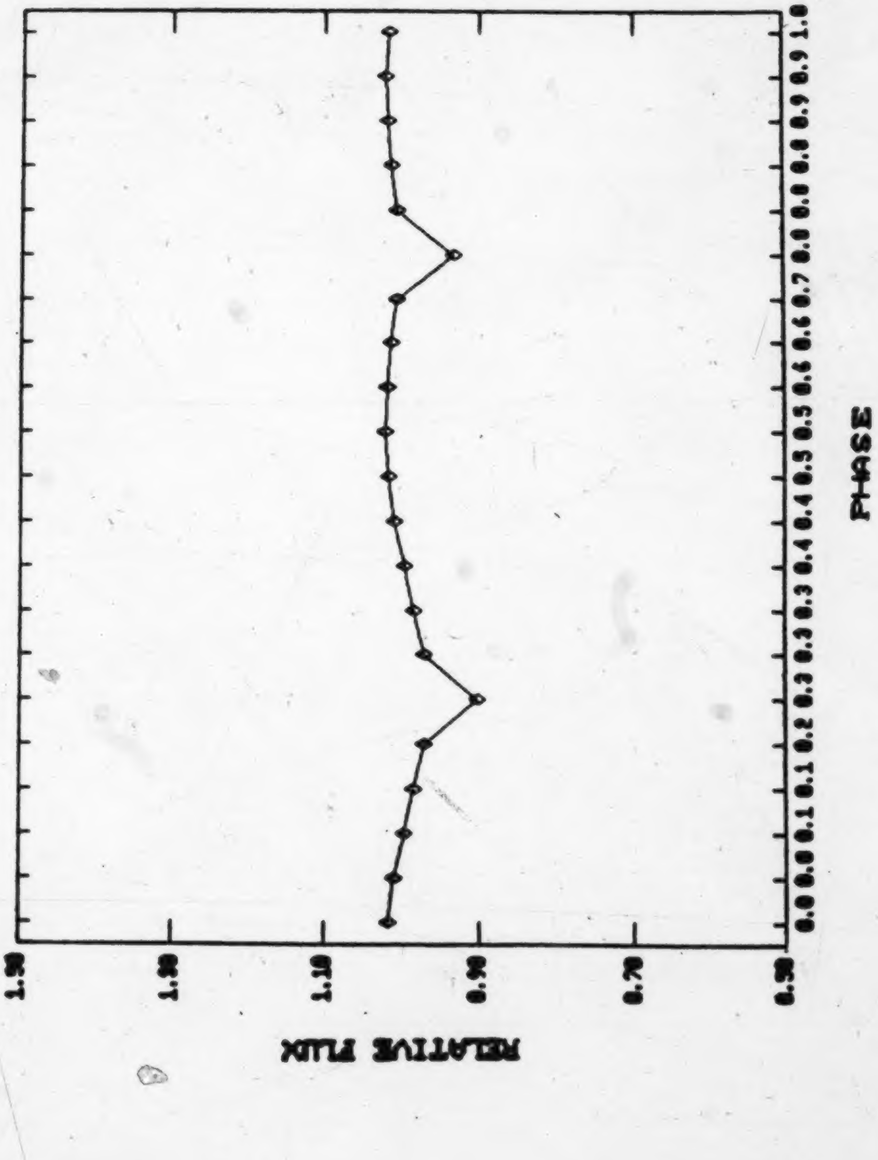
7.4.3. Trial Two

The second trial for HD219634 was based on a reexamination of the V photometry. After some 10-15 trials with LC it was decided that a new starting model with $i=70.0^{o}$,$\Omega_1=4.2$, and $\Omega_2=3.000$ would be employed with all other parameters the same as the DC trial one (Table VI). Fig. 7.4 illustrates the corresponding LC solution. Obviously, from the shape of the model it is more representative of the data than the original starting solution.

In running the DC solutions it was decided that the correlation between the potentials and inclination were too high to retain them in the same set. Thus, a new break down of subsets given by

Fig.7.4. Starting Solution for HD219634

(Trial Two).



$$\begin{array}{lll} \text{A)} & T_1 & \multirow{2}{*}{\}} \\ & \Omega_2 & \\ \text{B)} & \Omega 1 & \multirow{2}{*}{\}} \\ & L_2 & \\ \text{C)} & T_2 & \\ \text{D)} & L_1 & \\ \text{E)} & \Omega i & \end{array}$$

was employed. Table VII lists the corrections generated by each set.

After seven iterations all parameters but L_1 and L_2 had converged i.e. the suggested corrections were less than the probable error. Furthermore, the corrections for L_1 and L_2 were diminishing in magnitude after each successive correction. All indications were for a converging solution.

However, continued application of the corrections to L_1 and L_2 as suggested by DC lead to the solution being overshot. The reason for the continued corrections was due to the inadvertent setting of IPB = 1. Thus, DC was trying to fit L_1 and L_2 to the corresponding temperatures without any information on how they were related. Fig. 7.5 illustrates the solution after the first seven iterations. From fig. 7.5 we may make two immediate observations:

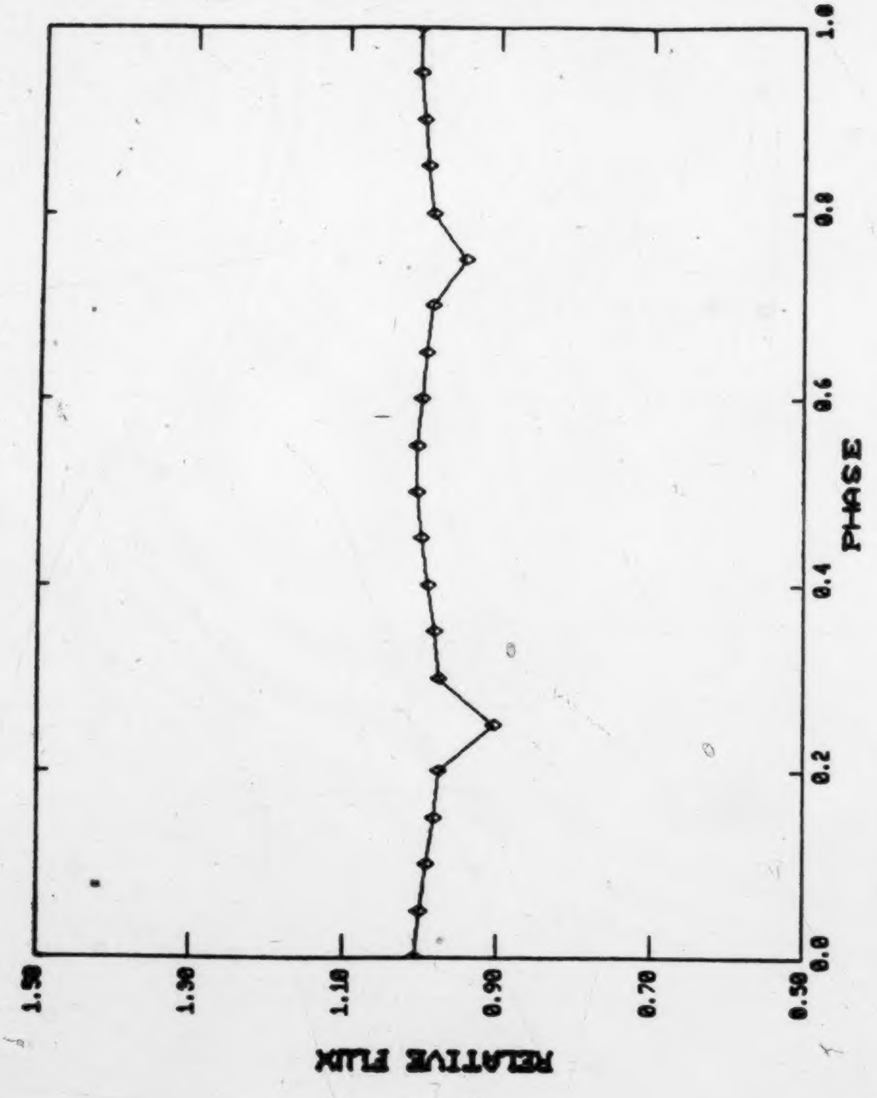
i) The secondary eclipse is underestimated in depth and,

ii) The primary eclipse is systematically underestimated in its flux variation.

Table VII Corrections for Trial Two.

Iteration	T_1	Ω_2	Ω_1	L_2	T_2	L_1	i
1	-.0608	0.0716	0.2163	-.0101	0.1868	-.0071	0.3224
2	0.1930	-.0238	0.2243	-.0086	0.2159	-.0064	0.3509
3	0.1995	-.0257	0.2546	-.0073	0.2098	-.0055	0.4103
4	0.1779	-.0099	0.1124	-.0051	0.1047	-.0041	0.1057
5	0.0847	-.0079	0.1107	-.0044	0.0764	-.0036	0.1030
6	0.0813	/	0.0622	-.0038	/	-.0036	0.1043
7	/	/	0.6044	-.0035	/	-.0032	0.8196

Fig.7.5. End of Seventh Iteration (Trial Two).



7.4.4. Trial Three

After a discussion with Wilson the problem of the IPB control parameter became obvious. A second suggestion made by Wilson was to fix the temperature of one of the stars. This would allow one to determine a solution as a relative measure. Since the best spectroscopic data was for the primary, it was decided this would be fixed at T_1=30000 K. Furthermore, one only varies one luminosity, L_1 in this scheme. Table VIII lists the new model parameters for HD219634.

The introduction of l_3, third light into the solution was suggested by Wilson to check whether there were any extraneous sources of light (dust, etc.) which may be contributing to the light. Parameter subsets were now:

A) $\left.\begin{array}{l}\Omega_2\\ L_1\end{array}\right\}$

B) $\left.\begin{array}{l}T_2\\ \Omega_1\end{array}\right\}$

C) $\left.\begin{array}{l}i\\ l_3\end{array}\right\}$

Convergence was achieved very quickly after four iterations, which suggested this strategy was correct. Table IX lists the final set of corrections, probable errors and residuals. The final parameters for HD219634 from V photometry are listed in Table X while Fig.7.6. illustrates the suggested light curve solution as plotted by LC.

Table VIII Starting Parameters for HD219634

(Trial Three)

Geometric Parameters	
$a = 21.3R_\odot$	$i = 71.282^\circ$ (varied)
$P = 2.3913^d$	$\omega = 222.5^\circ$
Physical Parameters	
$q = .3509$ (fixed)	
$\Omega_1 = 5.2238$ (varied)	$\Omega_2 = 2.9267$ (varied)
$L_1 = 7.1197$ (varied)	$L_2 = 5.4467$ (varied)
$x_1 = 0.6500$ (fixed)	$x_2 = 0.6500$ (fixed)
$T_1 = 3.0000$ (fixed)	$T_2 = 2.0000$(varied)

Table IX Corrections, Probable Errors, and Residuals for Third Trial

Iteration	Parameter	Corrections	P.E.'s	Residuals	
				Input	Predicted
	Ω_2	0.0578	0.0090	2.5450	0.0653
	L_1	0.0500	0.0023		
1	T_2	0.2193	0.1721	4.1300	0.0636
	Ω_1	0.0816	0.0338		
	i	0.0473	0.0847	0.0616	0.0614
	l_3	-.0031	0.0006		
	Ω_2	-.0007	-.0023	0.0706	0.0620
	L_1	-.0022	0.0026		
2	T_2	-.0161	0.1664	0.0706	0.0617
	Ω_1	0.0830	0.0288		
	i	0.0654	0.0843	0.0632	0.0614
	l_3	0.0017	0.0006		
	Ω_2	-.0014	0.0084	0.0616	0.0615
	L_1	-.0067	0.0022		
3	T_2	-.0038	0.0166	0.0616	0.0615
	Ω_1	0.0047	0.0290		
	i	0.0654	0.0847	0.0616	0.0614
	l_3	-.0001	0.0006		
	Ω_2	0.0058	0.0084	0.0610	0.0609
	L_1	0.0015	0.0022		
4	T_2	-.0151	0.0192	0.0610	0.0609
	Ω_1	0.2617	0.0338		
	i	0.0228	0.0883	0.0616	0.0614
	l_3	0.0001	0.0006		

Table X Final Parameters for HD219634

(V Photometry)

Geometric Parameters

$a = 21.3R_\odot$

$i = 71.282 \pm .086^\circ$

$P = 2.3913^d$

$\omega = 222.5^\circ$

Physical Parameters

$q = .3509$ (fixed)

$l_3 = 0.0000 \pm .0006$

$\Omega_1 = 5.2816 \pm .0338$

$\Omega_2 = 3.0083 \pm .0084$

$L_1 = 7.4964 \pm .0007$

$L_2 = 4.9264$ (fixed)

$x_1 = 0.6500$ (fixed)

$x_2 = 0.6500$ (fixed)

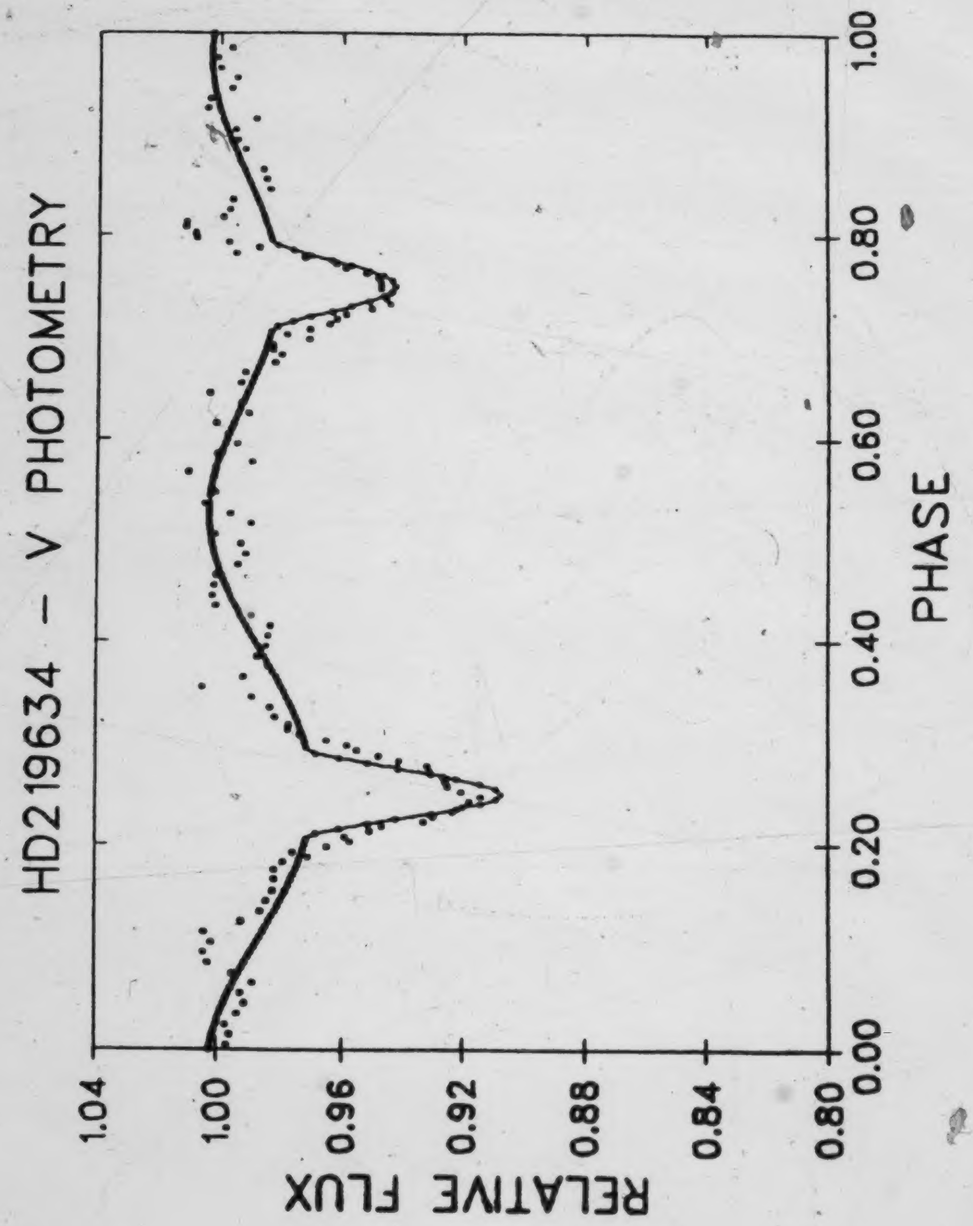
$T_1 = 3.0000$ (fixed)

$T_2 = 2.2003 \pm .0192$

Component	R Pole	R Point	R Side	R Back
1	.20252	.20475	.20367	.20448
2	.20589	.22096	.20945	.21756

Fig.7.6. Final Solution for HD219634.

V Photometry.



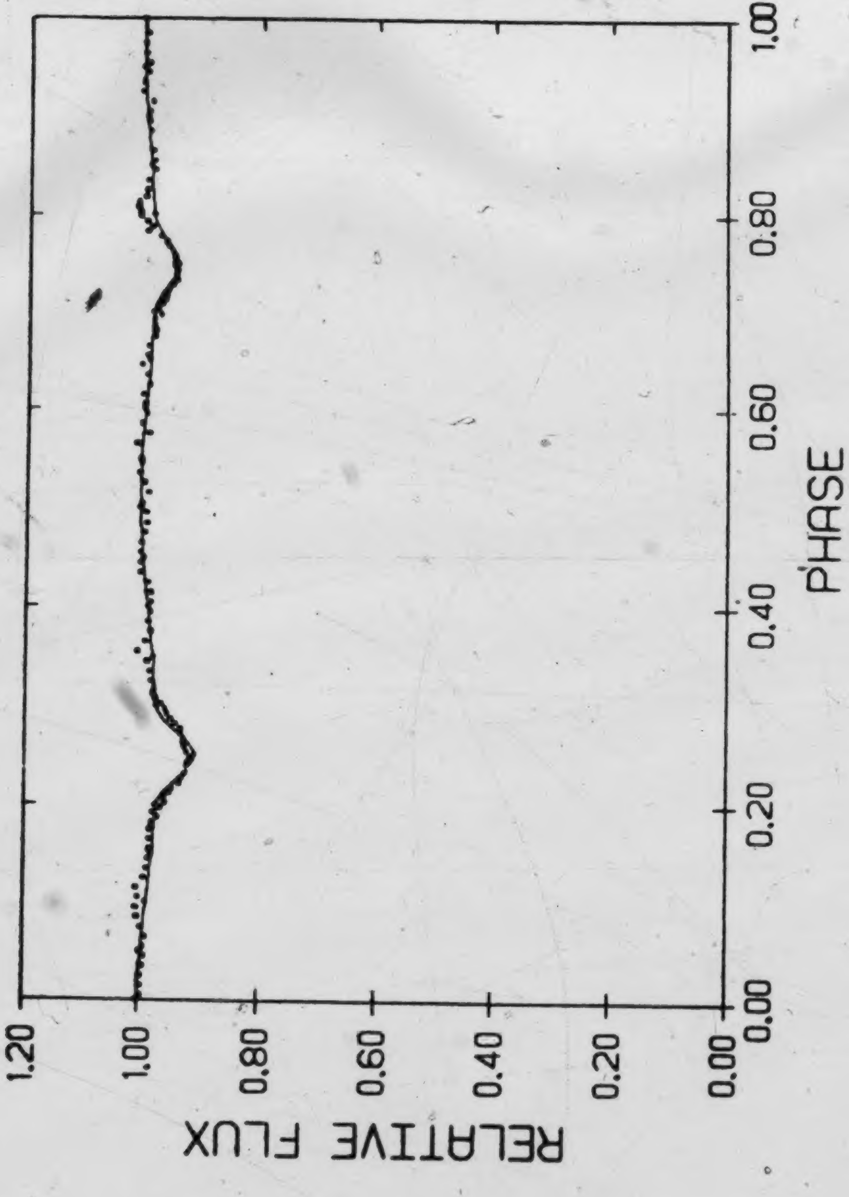
7.5. Interpretation of the Results

Central to the problem of analysis of HD219634 is the fact that the eclipses are partial . This leads to difficulty in constraining model parameters and ultimately obtaining a unique solution.

From the solution for HD219634 (see Fig.7.6.) we can see that the fit is far from ideal. The first impression of it is that the model curve fits the data fairly well in general shape but perhaps systematically sits above the data. If we consider the amplitude of this system is very small (typically, systems analyzed with the WD code have an amplitude of 1.0 in relative flux) then the fit to the data is reasonable (See Fig.7.7.).

A feature of the solution is the close fit to the depths of both the primary and secondary eclipses. This strongly suggests that the temperatures of the two components of the system are accurate. However, this is with the provisol that the estimations of our luminosities are correct. This is far from certain for partial eclipses and has been estimated from the relative depths of the eclipses. Furthermore, exactly where to measure this depth from is far from certain i.e. from the shoulders or from some midpoint of out-of-eclipse variation? The whole question of luminosity is far from settled for this system.

Fig.7.7. Fit to V Photometry for HD219634 on a coarser scale.



A strong indicator that the temperature differential is reasonable is the disparity in the height of the shoulders of the two eclipses. The increased luminosity of the secondary just outside of eclipse is a measure of the "reflection" or "heating" effect.

One feature of particular interest is the increased luminosity after the secondary eclipse (phase .78-.82). The solution suggests that the primary is close to filling its critical Roche lobe in which case it is possible that some material is being ejected through the inner Lagrangian point, L_1, from the secondary to the primary (see Fig.7.8. for a simple illustration of possible geometry). If indeed this is the case we are seeing the gas stream approaching us at this phase.

Unfortunately, the amplitude of the possible gas stream variation is so small that it is not possible to dismiss it as noise. However, comparison of the three light curves reveals this phenomenon only in the V filter (Fig.7.1.) which would correspond to a relatively cooler gas stream i.e. one originating at the cooler secondary. Closer examination of the primary eclipse (when the cooler secondary is between the observer and the hotter primary) shows a definite decrease in noise. This may be explained as the disappearance of the secondary's gas stream or lobe behind the body of the secondary.

Assuming the temperatures are correct we have a binary system whose massive component is a B0V star with a B2V companion (cf. Underhill and Doazan, p.54-57). The theoretical values of mass and radius for such stars are

Fig.7.8. Representations of HD219634 at phases: 0.25 (top) showing eclipse and 0.83 (bottom) showing secondary emerging from eclipse and gas stream visible.

B0V $16.8 \pm 4.3 M_\odot$

B2V $10.3 \pm 3.5 M_\odot$ $6.0 \pm 0.2 R_\odot$

(ibid, p.35-37). Comparing these values to the ones obtained by the WD code we note that the observed mass ratio ($q = 0.35$) falls within the rather large theoretical limits ($q = .32$-$.92$). Comparison of the observed values listed in Table X and assuming a mean radii we have

$$R_1 = .2039 \times 21.3 R_\odot = 4.3 R_\odot$$

$$R_2 = .2135 \times 21.3 R_\odot = 4.5 R_\odot$$

Hence, our observed and theoretical radii appear to be in close agreement. Our observed values may seem slightly small but certainly to within our present modelling they appear reasonable.

In the case of HD219634 where the secondary is just about to fill its Roche lobe we probably have a detached (almost semi-detached) system. Some values for detached systems are listed in Table XI (in particular note V PUP and μ Sco). We can see that the values for our radii are certainly not unusual.

As one final observation we may note that a system where the less massive component is in a semi-detached state is normally thought of as having already undergone mass transfer from the more massive to less massive component. If this is the case then what we are seeing as the primary component could be the "core" of the primary which may be overluminous for its mass.

Figure withheld due to copyright restriction.

For the final step in the thesis an analysis of the B photometry was performed. Fig.7.9. illustrates the fit to the B light curve and Table XII lists the corresponding solution. There is strong agreement between the B photometry and the V photometry.

7.6. Conclusion

From the preliminary analysis of the V and B photometry for HD219634 we may conclude that this star is a detached system (almost semi-detached) comprised of early B-type components (B0-B5). A check of the solution may possibly be obtained by solving the B and V photometry simultaneously with the radial velocity data. A disparity in the spectroscopic data, if it exists, may be detected at this time.

Table XII Final Parameters for HD219634

(B Photometry)

Geometric Parameters

$a = 21.3R_\odot$

$i = 71.582 \pm .085^{o}$

$P = 2.3913^{d}$

$\omega = 222.5^{o}$

Physical Parameters

$q = .3509$ (fixed)

$l_3 = 0.0009 \pm .0000$

$\Omega_1 = 5.4355 \pm .0356$

$\Omega_2 = 3.0312 \pm .0093$

$L_1 = 7.5183 \pm .0007$

$L_2 = 5.1197$(fixed)

$x_1 = 0.6500$ (fixed)

$x_2 = 0.6500$ (fixed)

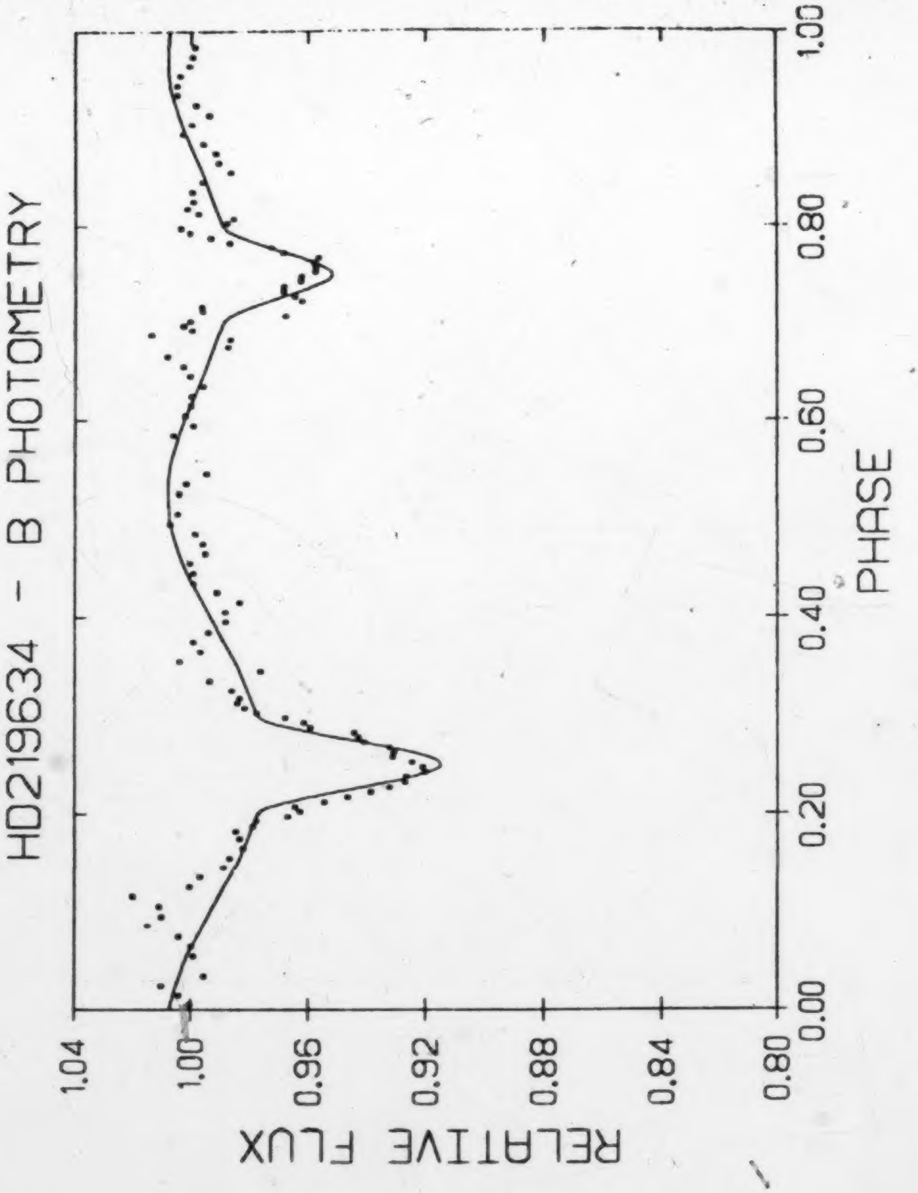
$T_1 = 3.0000$ (fixed)

$T_2 = 2.2289 \pm .0178$

Component	R Pole	R Point	R Side	R Back
1	.19641	.19838	.19744	.19815
2	.20322	.21736	.20659	.21425

Fig.7.9. Final Solution for HD219634.

B Photometry.



CHAPTER EIGHT

Conclusion

From the analysis of HD219634 in B and V photometry we can conclude that HD219634 is a detached system with its secondary component almost filling its critical Roche lobe. A combination of photometric and spectroscopic data in a solution must still be performed. This may suggest a small variation in the spectroscopic mass ratio (which did not consider ellipticity effects).

An explanation of the current state of HD219634 may well be as suggested in this thesis i.e. a primary which has already transferred mass to the secondary. However, this is not the only possibility. Hube has noted (private correspondence) that it is more likely that the present secondary was at one time the more massive component and we are seeing the final stages of the initial mass transfer. Without further information the two possiblities remain.

One final comment regarding the solution is a reiteration of the difficulty in finding a unique solution for systems displaying only partial eclipses. In the case of HD219634 this is compounded by the fact that the amplitude of light variation is so small.

We can see from the analysis of HD219634 that the WD code is extremely useful in analyzing close binary systems. However, it is also a complicated program which requires some experience and intuition to be effective.

A strategy of using the WD code in solving photometric/spectroscopic data is as follows:

i) Determine a preliminary model from theoretical models.

ii) Test with LC and make a preliminary solution and check on the gross affects of parameter interdependence.

iii) Run through DC to obtain a final solution paying special attention to:

a) Correlation within subsets.

b) Step sizes.

BIBLIOGRAPHY

Aitken, R.G., 1935, "The Binary Stars.", McGraw-Hill, New York.

Anton, H., 1981, "Elementary Linear Algebra.", John Wiley & Sons.

Binnendijk, L., 1960, "Properties of Double Stars.", Univ. of Pennsylvania Press.

Bowers, R.L., and Deeming, T., 1984, "Astrophysics I & II.", Jones and Bartlett.

Brancewicz, H., 1982, in "Ejection and Accretion of Matter in Binary Systems.", pp. 109-110.

Carbon, P.F., and Gingerich, O., in "Theory and Observation of Normal Stellar Atmospheres.", O. Gingerich editor, M.I.T. Press, pp. 377-400.

Chanan, G.A., Middleditch, J. and Nelson, J.E., 1976, Astrophys. J., **208**, 512.

Chandrasekhar, S., 1939, "An Introduction to the Study of Stellar Structure.", Univ. of Chicago Press.

Clayton, D.D., 1983, "Principles of Stellar Evolution and Nucleosynthesis.", Univ. of Chicago Press.

Collins II, G.W., 1989, "The Fundamentals of Stellar Astrophysics.", W.H. Freeman and Company.

Cooper, D., and Clancey, M., 1982, "Oh! Pascal!", W.W. Norton & Co.

Fracastoro, M.G., 1972, "An Atlas of Light Curves of Eclipsing Binaries.", Turin.

Gulliver, A.F., D.P.Hube, and G.Hill, 1985, Astron. Astrophys, 151, 254.

Gulliver, A.F.,D.P. Hube, and A. Lowe, 1982, I.A.U. Commision 27, No.2146.

Halbwachs, J.L., 1982, in "Binary and Multiple Stars as Tracers of Stellar Evolution.", Z. Kopal and J. Rahe editors, (Dordrecht:Reidel), pp. 199-204.

Hall, D.A., and Hardie, R.H., 1969, Publ. A.S.P., **81**, 754.

Harper, W.E., Pearce, J.A., Petrie, R.M., and McKellar, A., 1935, J.R.A.S., Canada, **29**, 411.

Hensler, G., 1982, in "Binary and Multiple Stars as Tracers of Stellar Evolution.", Z. Kopal and J. Rahe editors, (Dordrecht:Reidel), pp. 219.

Hill, G., and Hutchings, J.B., 1970, Astrophys. J., **162**, 265.

Holmgren, D., 1984, "Analysis of Eclipsing Binary Light Curves.", M.Sc. Thesis, Univ. of Alberta.

Hutchings, J.B., 1976, in "Structure and Evolution of Close Binary Systems.", P. Eggleton, S. Mitton and J. Whelan editors, (Dordrecht:Reidel), pp. 9-18.

Hynek, J.A., 1951, in "Astrophysics.", J.A. Hynek editor, McGraw-Hill, pp. 448-478.

Irwin, J.B., 1962, in "Astronomical Techniques (Stars and Stellar Systems II).", W.A. Hiltner editor, Chicago University Press, Chicago, pp. 584-626.

Johnston, R.L., 1982, "Numerical Methods.", John Wiley & Sons.

Kopal, Z., 1959, "Close Binary Systems.", John Wiley & Sons.

Kopal, Z., 1978, "Dynamics of Close Binary Systems.", (Dordrecht:Reidel).

Kopal, Z., 1979, "Language of the Stars.", (Dordrecht:Reidel).

Lucy, L.B., 1968, Astrophys. J., **153**, 877.

Markworth, N.L., 1978, Mon. Not. R. Astro. Soc., **187**, 699.

Markworth, N.L., and Michaels, E.J. Jr., 1982, Publ. Astron. Soc. Pacific, **94**, 350.

Martynov, D.Ya., 1971, "Eclipsing Systems with Deformed Components."

Missana, M., 1987, Astrophys. and Space Sc., **136**, 167.

Mochnacki, S.W., 1984, Astrophys, J. Suppl., **55**, 551.

Pecker, J.C., 1973, in "Problems of Calibration of Absolute Magnitudes and Temperatures of Stars.", B. Hauck and B.E. Westerlund editors, (Dordrecht:Reidel), pp. 173-221.

Petrie, R.M., 1962, in "Astronomical Techniques (Stars and Stellar Systems II).", W.A. Hiltner editor, Chicago University Press, Chicago, pp. 560-581.

Pierce, N.L., 1951, in "Astrophysics.", J.A. Hynek editor, McGraw-Hill, pp. 479-494.

Piotrowski, S.L., Rucinski, S.M., and Semeniuk, I., 1974, ACTA Astronomica, **24**, 389.

Plavec, M.J., 1964, BAC, **15**, No. 5.

Plavec, M.J., 1982, in "Binary and Multiple Stars as Tracers of Stellar Evolution.", Z. Kopal and J. Rahe editors, (Dordrecht:Reidel), pp. 159-182.

Popper, D.M., 1969, in "Theory and Observation of Normal Stellar Atmospheres.", O. Gingerich editor, M.I.T. Press, pp. 31-34.

Popper, D.M., 1973, Astrophys. J., **188**, 559.

Popper, D.M., 1974, Astrophys. J., **220**, L11.

Ralston, A., 1965, "A First Course in Numerical Analysis.", McGraw-Hill.

Rees, M.J., 1976, in "Structure and Evolution of Close Binary Systems.", P. Eggleton, S. Mitton and J. Whelan editors, (Dordrecht:Reidel), pp. 225-236.

Rule, W., 1983, "Fortran 77:A Practical Approach.", P.W.S. Publ.

Russell, H.N., 1912a, Astrophys. J., **35**, 315.

Russell, H.N., 1912b, Astrophys. J., **36**, 54.

Russell, H.N., and Merrill, J.E., 1952, Princeton Contri. No. 26.

Schwarzschild, M., 1957, "Structure and Evolution of the Stars.", Dover.

Tremko, J., 1982, in "Ejection and Accretion of Matter in Binary Systems.", J. Tremko editor, Slavek Academy of Sciences, pp. 39-56.

Underhill, A.B., and Doazan, V., 1982, "B Stars.", NASA.

Underhill, A.B., 1966, "The Early Type Stars.", (Dordrecht:Reidel)

Ureche, V., 1976, in "Structure and Evolution of Close Binary Systems.", P. Eggleton, S. Mitton and J. Whelan editors, (Dordrecht:Reidel), pp. 351-364.

Van de Kamp, P., 1967, "Principles of Astronomy.", W.H. Freeman and Co.

van Gent, R.H., 1989, "Studies on Interacting Binary Stellar Systems."

Wade, R.A., and Rucinski, S.M., 1985, Astron, Astrophys. Suppl. Ser., **60**, 471.

Wilson, R.E., 1979, Astrophys. J., **234**, 1054.

Wilson, R.E., 1970, Publ. of Astron. Soc. of the Pacific, **82**, 815.

Wilson, R.E., and Biermann, P., 1976, Astron. & Astrophys., **48**, 349.

Wilson, R.E. and De Luccia, M.R., 1972, Astrophys. J., **177**, 191.

Wilson, R.E., and Devinney, E.J., 1971, Astrophys. J., **166**. 605.

Wilson, R.E. and Devinney, E.J., 1973, Astrophys. J., **182**, 539.

Wilson, R.E. and Leung, K.C., 1976, Astrophys. J., **211**, 853.

Wilson, R.E. and Rafeit, J.B., 1980, Astron. Astrophys. Suppl. Ser., **42**, 195.

Wilson, R.E. and Sabatino, S., 1976, Astrophys. J., **203**, 182.

Wood, F.B., 1963, in "Basic Astronomical Data.", K.A. Strand editor, Univ. of Chicago Press, pp. 370-382.

APPENDIX A

The Wilson-Devinney Code

The following is a listing of those subroutines changed in the WD code enabling it to run in an "AT" environment. Compilation was performed using Microsoft FORTRAN (4.1) while the AT was an EPSILON 1600 with a 10 Mhz coprocessor.

All subroutines and drivers were compiled via

```
FL 'SUBROUTINE NAME.FOR'
```

A designation $LARGE at the beginning of each program specified large memory allocations.

All subroutines but not the drivers were added to a library WD.LIB via

```
LIB '+SUBROUTINE NAME'
```

To execute the program it is first necessary to link 'LCDRIVE' or 'DCDRIVE' to 'WD.LIB'. This is done by the command

```
LINK '_DRIVE' /ST:40000
```

The ST designation sets the stack size for the driver. An increase in the number of grid elements will possibly also require an increase in stack size (max. 65,636). Finally, to run the program simply type 'LCDRIVE' or 'DCDRIVE' as the case may be.

A.1. Subroutine LC : This subroutine generates synthetic light/radial velocity curves based upon input model parameters. It also generates a model of the physical dimensions of the component stars.

Strategy: As a simple algorithm we have the following :

i) Input model parameters.

ii) Generate initial grid elements.

iii) Calculate values of surface gravity, brightness, etc. normalized to an arbitrary point in the light curve.

iv) Calculate variation in flux/radial velocity at given phases normalized to above point.

v) Print values of light/radial velocity curve at each phase.

vi) Calculate dimensions of stars at periastron.

vii) Print values of radii, etc. at periastron.

```
$LARGE
      SUBROUTINE LC(RV, GRX, GRY, GRZ, RVQ, GRXQ, GRYQ, GRZQ,
SLUMP1,
     1        SLUMP2, MMSAVE, THETA, RHO, AA, BB, FR1, FR2, HLD,
     2        SNTH, CSTH, SNFI, CSFI, TLD)
C    VERSION OF 4/27/83. DESTROY EARLIER VERSIONS.
C
C    TO PRINT VELOCITIES IN KM/SEC, SET VUNIT=1.
C    TO PRINT NORMALIZED VELOCITIES IN SAME COLUMNS, SET
VUNIT
C    EQUATION TO DESIRED VELOCITY UNIT IN KM/SEC.
C
C      PARAMETER PSHIFT IS DEFINED AS THE PHASE AT WHICH
PRIMARY
C       CONJUNCTION (STAR 1 AWAY FROM OBSERVER) WOULD
OCCUR IF THE
C     LONGITUDE OF PERIASTRON WERE 90 DEGREES. SINCE THE
NOMINAL
C    VALUE OF THIS QUANTITY IS ZERO, PSHIFT MAY BE USED TO
C    INTRODUCE AN ARBITRARY PHASE SHIFT.
C
      DOUBLE PRECISION RES, DRR, ANS, DUMM
      DIMENSION RAD(2,4), DRDO(2,4), XLM(4), SQNU(4), RF(2), RMA(2),
     1        PO(2), XL1(2), OMEGA(2),
     7        GRZQ(1),
     6        RV(1), GRX(1), GRY(1), GRZ(1), RVQ(1), GRXQ(1), GRYQ(1),
     1        SLUMP1(1), SLUMP2(1), MMSAVE(1), THETA(1), RHO(1),
     2        AA(1), BB(1), FR2(1), HLD(1), SNTH(1), CSTH(1), SNFI(1),
     3        CSFI(1), TLD(1), FR1(1),
     4        DRR(4), RES(2), LX(2), MX(2), ANS(2), XLAT(2,100),
     5        XLONG(2,100)
```

```
      COMMON /FLVAR/ PER, PSHIFT, DP, DS, EF, EFC, ECOS, PERR,
PHPER,
    1       PCONJ, PHPERI, VSUM1, VSUM2, VRA1, VRA2, VKM1, VKM2,
VUNIT,
    2       VFVU
     COMMON /DPDX/ DPDX1, DPDX2, PHSV, PCSV
    COMMON /ECCEN/ E, A, PERIOD, VGA, SINI, VF, VFAC, VGAM, VOL1,

    1       VOL2, IFC
     COMMON /KFAC/ KFF1, KFF2
     COMMON /INVAR/ KH, IPB
     COMMON /SPOTS/ SINLAT(2,100), COSLAT(2,100), SINLNG(2,100),

    1       COSLNG(2,100), RADSP(2,100), TEMSP(2,100)
     COMMON /NSPT/ NSP1, NSP2, IFAT1, IFAT2
    DATA XLM(1), XLM(2), XLM(3), XLM(4), SQNU(1), SQNU(2), SQNU(3),

    1     SQNU(4) /0., 0., -1., 1., 0., 1., 1., 1./
   10 FORMAT (' PHASE   LITE 1  LITE 2  TOT LT  NRM LT    DIST   MAG+K

    1 VR1    VR2    DV1    DV2  R1POL  R1PT R1SID R1BAK R2POL  R2PT
R2S
    2ID R2BAK')
   20 FORMAT (' PHASE    LIGHT 1    LIGHT 2    (1+2+3)  NORM LITE     DI
    1ST    MAG+K     V RAD 1   V RAD 2    DEL V1     DEL V2  V KM/S 1  V
    2 KM/S 2')
   30 FORMAT (' PHASE    LIGHT 1    LIGHT 2    (1+2+3)  NORM LITE     DI
    1ST    MAG+K     V RAD 1   V RAD 2    DEL V1     DEL V2    VNORM 1
    2VNORM 2')
   40 FORMAT (1X, F6.4, 6F8.4, 4F7.3, 8(1X,F5.4))
   50 FORMAT (' WV LTH    L1      L2      X1      X2      EL3    M ZERO
    1FACTOR')
   60 FORMAT ('  ECC  LNG PER   S-M AXIS     F1       F2      V GAM  PSH
  1IFT  INCL    G1    G2   NSPOT1   NSPOT2')
 70 FORMAT ('    T1     T2    ALB 1   ALB 2   POT 1    POT 2    M2/
  1M1')
 80 FORMAT (3X, I2, 5X, I1, 6X, I1, 6X, I1, 4X, I2, 2X, I2, 3X, E10.5,
  1      1X, F5.4, 2X, F8.3, 2X, E14.4, 2X, F4.3, 3X, I3, 4X, I1)
 90 FORMAT (' DIMENSIONLESS RADIAL VELOCITIES CONTAIN FAC-
TOR P/(2PI*A)
  1')
 100 FORMAT (4X, 'GRID1/4    GRID2/4', 7X, 'POLAR SBR 1', 8X,
  1      'POLAR SBR 2', 7X, 'SURF. AREA 1', 7X, 'SURF. AREA 2', 6X,
  2      'PERI. PH.', 6X, 'CONJ. PH. ')
C 50 FORMAT(40H PRIMARY COMPONENT EXCEEDS CRITICAL LOBE)

C  51 FORMAT(42H SECONDARY COMPONENT EXCEEDS CRITICAL
LOBE)
C  43 FORMAT(1H1)
C  42 FORMAT(1H )
 110 FORMAT (' PRIMARY COMPONENT EXCEEDS CRITICAL LOBE')
  120 FORMAT (' SECONDARY COMPONENT EXCEEDS CRITICAL
LOBE')
 130 FORMAT (' ')
 140 FORMAT (' ')
 150 FORMAT (1X, 'COMPONENT', 3X, 'R POLE', 8X, 'DERIV', 8X, 'R
POINT',
  1      8X, 'DERIV', 9X, 'R SIDE', 8X, 'DERIV', 9X, 'R BACK', 8X,
  2      'DERIV')
 160 FORMAT (1X, F6.4, 1X, F7.5)
 170 FORMAT (F5.4, 1X, F6.2, 1X, F7.3, 1X, 2(F6.3,1X), F7.4, 1X, F5.4,
  1      1X, F6.3, 2(1X,F6.3))
 180 FORMAT (2(F6.4,1X), 2(F6.3,1X), 2(F7.4,1X), F8.5)
 190 FORMAT (1X, F6.4, 1X, 4(F9.6,2X), 2(F7.4,2X), 4F10.6, 2(1X,E10.4))
 200 FORMAT (I2, 1X, I1, 1X, I1, 1X, I1, 1X, I2, 1X, I2, 1X, E9.4, 1X,
    1       F5.4, 1X, F8.3, 1X, F4.3, 1X, I3, 1X, I1)
  210 FORMAT (F6.4, 2(1X,F7.4), 2(1X,F5.3), 1X, F6.4, 1X, F6.3, 1X,
    1       F6.4)
  220 FORMAT (1X, F5.4, 1X, F7.2, 2X, F9.4, 2(2X,F7.4), 2X, F7.4, 2X,
    1       F6.4, 2X, F6.3, 2(2X,F6.3), 2(5X,I3))
  230 FORMAT (1X, F6.4, 2(2X,F7.4), 2(2X,F5.3), 2X, F6.4, 2X, F6.3, 2X,
    1       F6.4)
  240 FORMAT (' PROGRAM SHOULD NOT BE USED IN MODE 1 OR 3
WITH NON-ZERO
    1ECCENTRICITY')
  250 FORMAT (' MODE  IPB  IFAT1  IFAT2  N1  N2   PERIOD    THE    V
    1 UNIT(KM/S)  V FAC     PHN   KPH  IFRAD')
  260 FORMAT (4X, I1, 8(2X,F12.5))
  270 FORMAT (1X, 2(I8,2X), F20.9, F19.9, F19.9, F18.9, F14.5, F15.5)
  280 FORMAT (2(2X,F6.4), 2(2X,F6.3), 2(2X,F7.4), 2X, F8.5)
  290 FORMAT (1X, I1, 4F12.3)
  300 FORMAT (3F7.2, F8.3)
  310 FORMAT (1X, 'STAR  CO-LATITUDE  LONGITUDE  SPOT RADIUS
TEMP. FACT
    1OR')
      KH = 17
      DTR = 1.745329E-2
       OPEN(UNIT=6,FILE='LC.OUT')
       OPEN(UNIT=5,FILE='WIL2.DAT')
      DO 490 IT = 1, 1000
       WRITE (6,130)
       READ (5,200) MODE, IPB, IFAT1, IFAT2, N1, N2, PERIOD, THE,
    1  VUNIT, PHN, KPH, IFRAD
       IF (MODE .GT. 91) STOP
       READ (5,170) E, PER, A, F1, F2, VGA, PSHIFT, XINCL, GR1, GR2
       READ (5,180) TAVH, TAVC, ALB1, ALB2, POTH, POTC, RM
      READ (5,210) WL, HLUM, CLUM, XH, XC, EL3, ZERO, FACTOR
      NSP1 = 0
      NSP2 = 0
      DO 330 KP = 1, 2
       DO 320 I = 1, 100
        READ (5,300) XLAT(KP,I), XLONG(KP,I), RADSP(KP,I),
    1    TEMSP(KP,I)
        IF (XLAT(KP,I) .GE. 200.) GO TO 330
        SINLAT(KP,I) = SIN(DTR*XLAT(KP,I))
        COSLAT(KP,I) = COS(DTR*XLAT(KP,I))
        SINLNG(KP,I) = SIN(DTR*XLONG(KP,I))
        COSLNG(KP,I) = COS(DTR*XLONG(KP,I))
        IF (KP .EQ. 1) NSP1 = NSP1 + 1
  320   IF (KP .EQ. 2) NSP2 = NSP2 + 1
  330  CONTINUE
      NSTOT = NSP1 + NSP2
      NP1 = N1 + 1
      NP2 = N1 + N2 + 2
      RF(1) = F1 ** 2
      RF(2) = F2 ** 2
      RMA(1) = RM
      RMA(2) = 1. / RM
      PERR = DTR * PER
      CALL SINCOS(1, N1, N1, SNTH, CSTH, SNFI, CSFI, MMSAVE)
      CALL SINCOS(2, N2, N1, SNTH, CSTH, SNFI, CSFI, MMSAVE)
         CALL BB1(RV, GRX, GRY, GRZ, RVQ, GRXQ, GRYQ, GRZQ,
MMSAVE, FR1,
    1    FR2, HLD, RM, POTH, POTC, GR1, GR2, ALB1, ALB2, N1, N2, F1,
    2     F2, MOD, XINCL, THE, SNTH, CSTH, SNFI, CSFI, MODE)
      CALL ELLONE(F1, DP, RM, XL1(1), OMEGA(1), XLD, OMD)
      CALL ELLONE(F2, DP, RMA(2), XL1(2), OMEGA(2), XLD, OMD)
      OMEGA(2) = OMEGA(2) * RM + (1. - RM) * .5
      MOD46 = (MODE - 5) ** 2
      IF (MODE .GE. 5) OMEGA(2) = POTC
      IF (MOD46 .EQ. 1) OMEGA(1) = POTH
      IF (E .EQ. 0.) GO TO 340
      DAP = 1. + E
      P1AP = POTH - 2. * E * RM / (1. - E*E)
     CALL VOLUME(VOL1, RM, P1AP, DAP, F1, N1, N1, KFD, 1, RV, GRX,

     1      GRY, GRZ, RVQ, GRXQ, GRYQ, GRZQ, MMSAVE, FR1, FR2,
 HLD,
     2      SNTH, CSTH, SNFI, CSFI, SUMMD, SMD, 2)
      DPDX1 = (POTH - P1AP) * (1. - E*E) * .5 / E
      P2AP = POTC - 2. * E / (1. - E*E)
     CALL VOLUME(VOL2, RM, P2AP, DAP, F2, N2, N1, KFD, 2, RV, GRX,

     1      GRY, GRZ, RVQ, GRXQ, GRYQ, GRZQ, MMSAVE, FR1, FR2,
 HLD,
     2      SNTH, CSTH, SNFI, CSFI, SUMMD, SMD, 2)
      DPDX2 = (POTC - P2AP) * (1. - E*E) * .5 / E
 340  CONTINUE
      PHSV = POTH
      PCSV = POTC
      IF (E .EQ. 0.) GO TO 350
      IF (MOD .EQ. 1) WRITE (6,240)
 350  CONTINUE
        CALL BB2(RV, GRX, GRY, GRZ, RVQ, GRXQ, GRYQ, GRZQ,
 MMSAVE, FR1,
     1      FR2, HLD, SLUMP1, SLUMP2, THETA, RHO, AA, BB, POTH,
 POTC,
     2      N1, N2, KF1, KF2, F1, F2, D, HLUM, CLUM, XH, XC, GR1, GR2,
     3      WL, SM1, SM2, TPOLH, TPOLC, SBRH, SBRC, IFAT1, IFAT2,
 TAVH,
4     TAVC, ALB1, ALB2, PHN, RM, XINCL, HOT, COOL, SNTH, CSTH,
5      SNFI, CSFI, TLD, MODE)
  KFF1 = KF1
  KFF2 = KF2
  KH = 0
  IF (KF1 .EQ. 1) WRITE (6,110)
  IF (KF2 .EQ. 1) WRITE (6,120)
  IF ((KF1 + KF2) .GT. 0) WRITE (6,140)
  WRITE (6,250)
  WRITE (6,80) MODE, IPB, IFAT1, IFAT2, N1, N2, PERIOD, THE,
1 VUNIT, VFAC, PHN, KPH, IFRAD
  WRITE (6,140)
  WRITE (6,60)
  WRITE (6,220) E, PER, A, F1, F2, VGA, PSHIFT, XINCL, GR1, GR2,
1 NSP1, NSP2
  WRITE (6,140)
  WRITE (6,70)
  WRITE (6,280) TAVH, TAVC, ALB1, ALB2, PHSV, PCSV, RM
  WRITE (6,140)
  WRITE (6,50)
  WRITE (6,230) WL, HLUM, CLUM, XH, XC, EL3, ZERO, FACTOR
  WRITE (6,140)
  IF (NSTOT .GT. 0) WRITE (6,310)
  DO 370 KP = 1, 2
   IF ((NSP1 + KP - 1) .EQ. 0) GO TO 370
   IF ((NSP2 + (KP - 2)**2) .EQ. 0) GO TO 370
   NSPOT = NSP1
   IF (KP .EQ. 2) NSPOT = NSP2
   DO 360 I = 1, NSPOT
```

```
360  WRITE (6,290) KP, XLAT(KP,I), XLONG(KP,I), RADSP(KP,I),
  1  TEMSP(KP,I)
370  WRITE (6,140)
     WRITE (6,100)
     WRITE (6,270) MMSAVE(NP1), MMSAVE(NP2), SBRH, SBRC, SM1,
SM2,
  1 PHPERI, PCONJ
     WRITE (6,140)
     ALL = HOT + COOL + EL3
     IF (MODE .EQ. - 1) ALL = COOL + EL3
     IF (IFRAD .EQ. 0) GO TO 380
     WRITE (6,10)
     GO TO 390
380  IF (VUNIT .EQ. 1.) WRITE (6,20)
     IF (VUNIT .NE. 1.) WRITE (6,30)
390  CONTINUE
     LIM = 501
     IF (E .NE. 0.) LIM = 1001
     IF (NSTOT .GT. 0) LIM = 1001
     LL1 = MMSAVE(N1) + 1
     NPP2 = NP2 - 1
     LL2 = MMSAVE(NPP2) + 1
     LLL1 = MMSAVE(NP1)
     LLL2 = MMSAVE(NP2)
     LLLL1 = (LL1 + LLL1) / 2
     LLLL2 = (LL2 + LLL2) / 2
     POTH = PHSV
     POTC = PCSV
     PO(1) = POTH
     PO(2) = POTC
     DO 410 I = 1, LIM, KPH
```

```
        EYE = I - 1
        PHAS = .001 * EYE
           CALL BB2(RV, GRX, GRY, GRZ, RVQ, GRXQ, GRYQ, GRZQ,
MMSAVE,
    1         FR1, FR2, HLD, SLUMP1, SLUMP2, THETA, RHO, AA, BB,
POTH,
    2      POTC, N1, N2, KF1, KF2, F1, F2, D, HLUM, CLUM, XH, XC,
    3         GR1, GR2, WL, SM1, SM2, TPOLH, TPOLC, SBRH, SBRC,
IFAT1,
    4      IFAT2, TAVH, TAVC, ALB1, ALB2, PHAS, RM, XINCL, HOT,
    5      COOL, SNTH, CSTH, SNFI, CSFI, TLD, MODE)
        HTT = HOT
        IF (MODE .EQ. - 1) HTT = 0.
        TOTAL = HTT + COOL + EL3
        TOTALL = TOTAL / ALL
        TOT = TOTALL * FACTOR
        SMAGG = -1.085736 * ALOG(TOTALL) + ZERO
        IF (IFRAD .EQ. 0) GO TO 400
       WRITE (6,40) PHAS, HTT, COOL, TOTAL, TOT, D, SMAGG, VSUM1,

    1  VSUM2, VRA1, VRA2, RV(1), RV(LL1), RV(LLLL1), RV(LLL1),
    2  RVQ(1), RVQ(LL2), RVQ(LLLL2), RVQ(LLL2)
        GO TO 410
  400     WRITE (6,190) PHAS, HTT, COOL, TOTAL, TOT, D, SMAGG,
VSUM1,
    1  VSUM2, VRA1, VRA2, VKM1, VKM2
  410  CONTINUE
      POTH = PHSV
      POTC = PCSV
         CALL BB2(RV, GRX, GRY, GRZ, RVQ, GRXQ, GRYQ, GRZQ,
MMSAVE, FR1,
    1       FR2, HLD, SLUMP1, SLUMP2, THETA, RHO, AA, BB, POTH,
POTC,
     2    N1, N2, KF1, KF2, F1, F2, D, HLUM, CLUM, XH, XC, GR1, GR2,
     3     WL, SM1, SM2, TPOLH, TPOLC, SBRH, SBRC, IFAT1, IFAT2,
TAVH,
     4     TAVC, ALB1, ALB2, PHPERI, RM, XINCL, HOT, COOL, SNTH,
CSTH,
     5    SNFI, CSFI, TLD, MODE)
      WRITE (6,140)
      WRITE (6,150)
      WRITE (6,140)
      QF = 1.
      DO 470 II = 1, 2
       IF (II .EQ. 2) QF = 1. / RM
       DO 470 I = 1, 4
        RM1 = RMA(II) + 1.
        POT = PO(1)
        IF (II .EQ. 2) POT = PO(2) / RM + .5 * (RM - 1.) / RM
        IF (I .EQ. 1) R = 1. / POT
        KOUNT = 0
        DELR = 0.
 420    R = R + DELR
        KOUNT = KOUNT + 1
        IF (KOUNT .LT. 20) GO TO 440
 430    DOMR = -1.
        R = -1.
        GO TO 460
 440    RSQ = R * R
        PAR = DS - 2. * XLM(I) * R * DP + RSQ
        RPAR = SQRT(PAR)
        OM = 1. / R + RMA(II) * ((1./RPAR) - XLM(I)*R/DS) + RM1 * .
     1   5 * RSQ * SQNU(I) * RF(II)
        DOMR = 1. / (RF(II)*RM1*SQNU(I)*R - 1./RSQ - (RMA(II)*(R -
     1     XLM(I)*DP))/(PAR*RPAR) - RMA(II)*XLM(I)/DS)
           DELR = (POT - OM) * DOMR
           ABDELR = ABS(DELR)
           IF (ABDELR .GT. .00001) GO TO 420
           IF (R .GE. 1.) GO TO 430
           IF (I .LT. 4) GO TO 460
           IF (PO(II) - OMEGA(II)) 430, 450, 460
  450        R = XL1(II)
           DOMR = 0.
  460        DRDO(II,I) = DOMR * QF
           RAD(II,I) = R
  470    CONTINUE
       DO 480 II = 1, 2
  480    WRITE (6,260) II, RAD(II,1), DRDO(II,1), RAD(II,4), DRDO(II,4),
     1  RAD(II,2), DRDO(II,2), RAD(II,3), DRDO(II,3)
       WRITE (6,140)
       WRITE (6,90)
  490 CONTINUE
     STOP
     END
```

A.2. Subroutine DC : This program calculates the least-squares solution for the light/radial velocity of a binary system by the method of diffential corrections.

Strategy : As a basic outline of our algorithm we have:

i) Input starting parameters.

ii) generate initial grid elements.

iii) Normalize solution to arbitrary phase.

iv) For each phase :

a) Calculate unperturbed flux.

b) Calculate perturbed flux.

c) Calculate derivatives.

d) Calculate (O-C)'s.

e) Form normal equations.

v) Solve system of normal equations :

a) Properly weight normal equations.

b) Use Gaussian elimination with pivoting to solve.

vi) Print out values as requested.

a) Input values.

b) Correlation coefficients.

c) Normal equations times inverse.

d) Check.

e) At the very least, one must output the corrections, the associated probable errors and sum of residuals squared (input and predicted). It is also possible to print the matrix x inverse, correlation coefficients, and normal equations.

f) Dimensions, etc. for the system.

vii) Save matrix of derivatives, etc. for later use in subsets if desired.

```
$LARGE
   SUBROUTINE DC(PHAS, FLUX, BR, OBS, OUT, PE, CN, CCL, LL, MM,
HLA,
   1 CLA, X1A, X2A, EL3A, WLA, NOISE, KNOBS, RV, GRX, GRY,
   2 GRZ, RVQ, GRXQ, GRYQ, GRZQ, SLUMP1, BL, SLUMP2,
   5 MMSAVL, THETA, RHO, AA,
   6 BB, SIGMA, WT, FR1, FR2, HOLD, HLD, CNN, SNTHH, CSTHH,
   7 SNFIH, CSFIH, SNTHL, CSTHL, SNFIL, CSFIL, TLDH, TLDL,
   8 V, CNOUT, MMSAVH)
C    VERSION OF DEC. 28, 1985. DESTROY EARLIER VERSIONS.
C
C      PARAMETER NUMBER 1 IS A, THE RELATIVE ORBITAL SEMI-
MAJOR AXIS, IF
C    SIMULTANEOUS LIGHT AND VELOCITY SOLUTIONS ARE BEING
DONE. HOWEVER,
C    IF ONLY VELOCITY CURVES ARE BEING SOLVED, PARAMETER 1
WILL
```

```
C     EFFECTIVELY BE A*SIN(I), PROVIDED AN INCLINATION OF 90
DEG. IS

C    ENTERED. CONCEIVABLY,IN SOME RARE SITUATIONS IT MAY BE
POSSIBLE TO

C     FIND A AND I SEPARATELY, USING ONLY VELOCITY CURVES.
THIS COULD BE

C    THE CASE IF THE VELOCITY PROXIMITY EFFECTS ARE IMPOR-
TANT.

C

C    ALTHOUGH THE LONGITUDE OF PERIASTRON IS ENTERED AND
PRINTED IN

C    DEGREES, THE COMPUTED CORRECTION IS IN RADIANS.

C

C     OTHER PROGRAM UNITS: ORBITAL S-M AXIS IN SOLAR RADII
(6.96E5 KM),

C    PERIOD IN DAYS, PHASE IN 2 PI RADIANS, SYSTEMIC VELOCITY
AND

C    THIRD LIGHT IN SAME UNITS AS VELOCITY AND LIGHT OBSER-
VATIONS,

C    INCLINATION IN DEGREES, TEMPERATURES IN 10000K.

C

   DIMENSION PHAS(1), FLUX(1), BR(1), OBS(1), OUT(1), PE(1), CN(1),

   1CCL(1), LL(1), MM(1), HLA(1), CLA(1), X1A(1), X2A(1),
   2EL3A(1), WLA(1), NOISE(1), KNOBS(1), RV(1), GRX(1),
   3GRY(1), GRZ(1), RVQ(1), GRXQ(1), GRYQ(1), GRZQ(1),
   4SLUMP1(1), BL(1), SLUMP2(1), THETA(1), RHO(1), AA(1),
   5BB(1), SIGMA(1), WT(1), FR1(1), FR2(1), SNTHH(1),
   6CSTHH(1), SNFIH(1), CSFIH(1), SNTHL(1), CSTHL(1),
   7SNFIL(1), CSFIL(1), HOLD(1), HLD(1), CNN(1), V(1),
   8CNOUT(1)
   DIMENSION TLDH(1), TLDL(1), MMSAVL(1), MMSAVH(1)
   DIMENSION DEL(22), KEEP(23), KEP(22), NSHIFT(23), LOW(22)
   DIMENSION XTHA(4), XFIA(4), CCA(40), CCB(1600), XLAT(2,100),
    1XLONG(2,100)
    CHARACTER ARAD(4)
     COMMON /ECCEN/ EC, A, PERIOD, VGA, SINI, VF, VFAC, VGAM,
VOL1,
   1    VOL2, IFC
    COMMON /DPDX/ DPDX1, DPDX2, POT1, POT2
     COMMON /FLVAR/ PER, PSH, DP, DSDUM, EF, EFC, ECOS, PERT,
PHPER,
     1    PCONJ, PHPERI, VSUM1, VSUM2, VRA1, VRA2, VKM1, VKM2,
VUNIT,
   2    VFVU
    COMMON /INVAR/ KH, IPB
    COMMON /INCON/ IFVC1, IFVC2, NLC
    COMMON /KFAC/ KFF1, KFF2
    COMMON /SPOTS/ SINLAT(2,100), COSLAT(2,100), SINLNG(2,100),

   1   COSLNG(2,100), RADSP(2,100), TEMSP(2,100)
    COMMON /NSPT/ NSP1, NSP2, IFAT1, IFAT2
    COMMON /CNTRL/ DL,MM1,MM2,MM3,MM4,DLIF(22)
    COMMON /VCOM/ SRV(0762),SGRX(0762),SGRY(0762),
   1    SGRZ(0762),SLMP1(0762)
    COMMON /VCOMQ/ SRVQ(0762),SGRXQ(0762),SGRYQ(0762),
   1    SGRZQ(0762),SLMP2(0762)
    COMMON /VCOML/ SRVL(0762),SGRXL(0762),SGRYL(0762),
   1    SGRZL(0762),SLMP1L(0762)
    COMMON /VCOMQL/ SRVQL(0762),SGRXQL(0762),SGRYQL(0762),
   1    SGRZQL(0762),SLMP2L(0762)
    DOUBLE PRECISION CN,CCL,OUT,D,CNN,V,ANSCH,CNOUT,ERR
C                                                          DATA
ARAD(1),ARAD(2),ARAD(3),ARAD(4)/4HPOLE,5HPOINT,4HSIDE,4HBAC
K/
  10 FORMAT (1X, 16(F8.5))
  20 FORMAT (1X, 18(F7.4))
```

```
 30 FORMAT (20A4)
 40 FORMAT (' SPECIFIED ECLIPSE DURATION INCONSISTENT WITH
OTHER PARAM
   1ETERS')
 50 FORMAT (1X, 22(F6.3))
 60 FORMAT (1X, 26(F5.2))
 70 FORMAT (2(1X,F35.25), 1X, E12.6)
 80 FORMAT (10(3X,E8.1))
 90 FORMAT (10(1X,E7.1))
100 FORMAT (1X, 22I1, 1X, I1, 1X, I1, 1X, I1)
110 FORMAT ('1')
120 FORMAT (' ')
130 FORMAT (3X, I2, 5X, I1, 5X, I1, 6X, I1, 4X, I2, 2X, I2, 2X, I2,
  1     2X, I2, 3X, E10.5, 3X, F5.4, 8X, F8.3, 5X, E9.4)
140 FORMAT (5(F6.4,F7.4,F3.1))
150 FORMAT (1X, 5(F7.4,F7.4,F4.1))
160 FORMAT (1X, F6.4, 2(1X,F7.4), 2(1X,F5.3), 1X, F6.4, 1X, I1, 1X,
  1     F6.4)
170 FORMAT (1X, F6.4, 2(1X,F7.4), 2(1X,F5.3), 1X, F6.4)
180 FORMAT (1X, 11F12.7)
190 FORMAT (1X, F11.7)
200 FORMAT (1X, 'SUM OF ABSOLUTE VALUES OF CHECKS IS', 1X,
D12.6)
210 FORMAT (7X, 'NORMAL EQUATIONS')
220 FORMAT (7X, 'CORRELATION COEFFICIENTS')
230 FORMAT (7X, 'NORMAL EQUATIONS TIMES INVERSE')
240 FORMAT (1X, 'CHECK OF COMPUTED DEPENDENT VARIABLES
FROM NORMAL QUA
   1T IONS')
250 FORMAT (7X, 'UNWEIGHTED OBSERVATIONAL EQUATIONS')
260 FORMAT (7X, 'WEIGHTED OBSERVATIONAL EQUATIONS')
270 FORMAT (33X, 'OBSERVATIONS')
```

```
 280 FORMAT ('  PHASE  V RAD  WT  PHASE  V RAD  WT  PHASE  V
RAD  WT  P
   1HASE  V RAD  WT  PHASE  V RAD  WT')
 290 FORMAT (' PHASE LIGHT WT PHASE LIGHT WT PHASE LIGHT
WT  P
   1HASE  LIGHT  WT  PHASE  LIGHT  WT')
 300 FORMAT (' PROGRAM SHOULD NOT BE USED IN MODE 1 OR 3
WITH NON-ZERO
   1ECCENTRICITY')
 310 FORMAT ('  PARAMETER CORRECTIONS')
 320 FORMAT ('  PROBABLE ERRORS')
   330  FORMAT  ('   SUM(W*RES**2)  FOR  INPUT  VALUES
SUM(W*RES**2)PRED
   1 ICTED     DETERMINANT')
 340 FORMAT (' WAVE L  L1  L2 X1    X2  3RD L   SIGMA')
 350 FORMAT (' WAVE L  L1  L2 X1    X2   SIGMA')
 360 FORMAT (' MODE   IPB  IFAT1  IFAT2  N1  N2  N1L  N2L   PERIOD

   1 TH E  V UNIT(KM/S)    V FAC')
 370 FORMAT (1X, F5.4, 1X, F7.2, 2X, F9.4, 2(2X,F7.4), 1X, F8.4, 2X,
   1     F6.4, 2X, F6.3, 2(2X,F6.3))
 380 FORMAT ('  ECC  LNG PERS-M AXIS     F1     F2    V GAM  PSHIF
   1T  INCL    G1    G2')
 390 FORMAT ('   DEL EC      DEL PER DEL F1    DEL F2     DEL PHS
   1  DEL INCL    DEL G1     DEL G2  DEL T1      DEL T2')
 400 FORMAT ('   DEL ALB1   DEL ALB2 DEL POT1   DEL POT2   DEL Q

   1  DEL L1   DEL L2     DEL X1 DEL X2')
 410 FORMAT (' ORDER OF PARAMETERS IS  A,E,PER,F1,F2,PHASE
SHIFT,VGAM,I
        1NCL,G1,G2,T1,T2,ALB1,ALB2,POT1,POT2,Q,L1,L2,X1,X2,3RD
LIGHT.')
 420 FORMAT (' ADJUSTMENT CONTROL INTEGERS; 1 SUPPRESSES
ADJUSTMENT, 0
```

```
  1ALLOWS ADJUSTMENT.')
430 FORMAT (I2, 1X, I1, 1X, I1, 1X, I1, 1X, 4(I2,1X), E9.4, 1X, F5.4,
  1    1X, F8.3)
440 FORMAT (F5.4, 1X, F6.2, 1X, F7.3, 1X, 2(F6.3,1X), F7.4, 1X, F5.4,
  1    1X, F6.3, 2(1X,F6.3))
450 FORMAT (2(F6.4,1X), 2(F6.3,1X), 2(F7.4,1X), F8.5)
460 FORMAT (I1, 1X, I1, 1X, I2, 3(1X,I1))
470 FORMAT ('    T1    T2   ALB 1   ALB 2   POT 1    POT 2    M2/
  1M1')
480 FORMAT (2(2X,F6.4), 2(2X,F6.3), 2(2X,F7.4), 2X, F8.5)
490 FORMAT ('   IFVC1   IFVC2   NLC KO   KDISK   ISYM')
500 FORMAT (6(4X,I3))
510 FORMAT (20X, 'RADII AND RELATED QUANTITIES (FROM INPUT)')
520 FORMAT (5X, 'DOM1/DQ', 5X, 'DOM2/DQ', 5X, 'OM1-Q CORR.', 5X,

  1    'OM2-Q CORR.', 5X, 'OM1 P.E.', 4X, 'OM2 P.E.', 4X,
  2    'Q  P.E.')
530 FORMAT (1X, 3F12.6, 4X, F12.6, 4X, 3F12.6)
540 FORMAT (' COMPONENT', 11X, 'R', 9X, 'DR/DOM', 8X, 'DR/DQ', 11X,

  1    'P.E.')
550 FORMAT (2X, I1, 1X, A6, 4F14.6)
560 FORMAT (1X, I1, 4F12.3)
570 FORMAT (3F7.2, 2F8.3)
580 FORMAT (1X, 'STAR  CO-LATITUDE  LONGITUDE  SPOT RADIUS
TEMP.FACTO
  1R ')
  ARAD(1) = 'POLE'
  ARAD(2) = 'POINT'
  ARAD(3) = 'SIDE'
  ARAD(4) = 'BACK'
  DTR = 1.745329E-2
```

```
      PI = 3.141593
      XTHA(1) = 0.
      XTHA(2) = .5 * PI
      XTHA(3) = .5 * PI
      XTHA(4) = .5 * PI
      XFIA(1) = 0.
      XFIA(2) = 0.
      XFIA(3) = .5 * PI
      XFIA(4) = PI
      IBEF = 0
      KFF1 = 0
      KFF2 = 0
      II = 0
      KH = 17
      NS = 1
      NI = 0
      NY = 0
      KNOBS(1) = 0
      OPEN(UNIT=6,FILE='DC.OUT')
      WRITE (6,110)
C
      CALL INPUT1(DEL,KEP,IFDER,IFM,IFR,KO,KDISK,
     1 ISYM,MODE,N1,N2,
     2 N1L,N2L,THE,E,F1,F2,
     3 PSHIFT,XINCL,GR1,GR2,TAVH,TAVC,
     4 ALB1,ALB2,PHSV,PCSV,RM)
      CALL SINCOS(1, N1, N1, SNTHH, CSTHH, SNFIH, CSFIH, MMSAVH)
      CALL SINCOS(2, N2, N1, SNTHH, CSTHH, SNFIH, CSFIH, MMSAVH)
      CALL SINCOS(1, N1L, N1L, SNTHL, CSTHL, SNFIL, CSFIL, MMSAVL)
      CALL SINCOS(2, N2L, N1L, SNTHL, CSTHL, SNFIL, CSFIL, MMSAVL)
```

```
C
      IS = ISYM + 1
      KEEP(23) = 0
      MM1 = N1 + 1
      MM2 = N1 + N2 + 2
      MM3 = N1L + 1
      MM4 = N1L + N2L + 2
      NVC = IFVC1 + IFVC2
      NLVC = NLC + NVC
      NVCP = NVC + 1
      IF (NVC .NE. 0) GO TO 590
      KEP(1) = 1
      KEP(7) = 1
  590 CONTINUE
      DO 600 I = 1, 22
  600 KEEP(I) = KEP(I)
      DO 610 I = 1, 22
  610 LOW(I) = 1
      LOW(2) = 0
      LOW(3) = 0
      LOW(4) = 0
      LOW(5) = 0
      LOW(6) = 0
      LOW(8) = 0
      LOW(15) = 0
      LOW(16) = 0
      LOW(17) = 0
      KOSQ = (KO - 2) * (KO - 2)
C
      CALL INPUT2(NVC,NVCP,NLVC,NSP1,NSP2,WLA,HLA,CLA,
    1 X1A,X2A,SIGMA,EL3A,NOISE,XLAT,XLONG)
C
      NSTOT = NSP1 + NSP2
      PERR = PER * DTR
      PERT = PERR
      EC = E
      PSH = PSHIFT
C
      CALL CALC1(RV,GRX,GRY,GRZ,RVQ,GRXQ,GRYQ,GRZQ,
    1 MMSAVH,FR1,FR2,HLD,RM,PHSV,PCSV,
    2 GR1,GR2,ALB1;ALB2,N1,N2,F1,F2,
    3 MOD,XINCL,THE,SNTHH,CSTHH,SNFIH,
    4 CSFIH,MODE)
C
      PHP = PHPER
      POTH = PHSV
      POTC = PCSV
      POT1 = PHSV
      POT2 = PCSV
      DO 690 I = 1, NLVC
C
  690 CALL BB2(RV, GRX, GRY, GRZ, RVQ, GRXQ, GRYQ, GRZQ,
MMSAVH, FR1,
    1 FR2, HLD, SLUMP1, SLUMP2, THETA, RHO, AA, BB, POTH, POTC,
N1,
    2 N2, KF1, KF2, F1, F2, D, HLA(I), CLA(I), X1A(I), X2A(I), GR1,
    3 GR2, WLA(I), SM1, SM2, TPOLH, TPOLC, SBRH, SBRC, IFAT1,
    4 IFAT2, TAVH, TAVC, ALB1, ALB2, PHP, RM, XINCL, HOT, COOL,
    5 SNTHH, CSTHH, SNFIH, CSFIH, TLDH, MODE)
C
      NCF = 1
      DEL(1) = 0.
      DEL(7) = 0.
      DEL(22) = 0.
C
     CALL OUTPT1(NVC,NVCP,NLVC,DEL,KEP,IFDER,IFM,IFR,
    1 KO,KDISK,ISYM,MODE,N1,N2,N1L,N2L,THE,
    2 PSHIFT,XINCL,GR1,GR2,TAVH,TAVC,
    3 X1A,X2A,SIGMA,EL3A,NOISE,
    4 ALB1,ALB2,PHSV,PCSV,RM,WLA,HLA,CLA,
    5 NSTOT,XLAT,XLONG)
C
      CALL INOUT1(NS,NVC,NLVC,PHAS,FLUX,
    1  WT,NOBS,KNOBS,LC1)
      DO 999 I=1,6
  999 CONTINUE
C
      CALL MATFOR(KEEP,MAT,KTR,NCOEFF,
    1  KNOBS(LC1))
C
      NMAT = MAT * KNOBS(LC1)
      NCOF = NCOEFF
      DO 790 J = 1, NCOF
  790 HOLD(J) = 0.
      IF (KOSQ .EQ. 1) GO TO 1170
      DO 800 IOBS = 1, NCOEFF
  800 OBS(IOBS) = 0.
      DO 1160 IB = 1, NLVC
C     VC1 = 0.
C     VC2 = 0.
C     ELIT = 0.
```

```
IF (IB .GT. NVC) ELIT = 1.
IF (IB .EQ. IFVC1) VC1 = 1.
IF (IB .EQ. (IFVC2*(1 + IFVC1))) VC2 = 1.
IST = KNOBS(IB) + 1
IB1 = IB + 1
ISP = KNOBS(IB1)
DO 1150 IX = IST, ISP
  DO 1140 KS = 1, 24
    KH = KS - 2
    IF (KS .LE. 2) GO TO 940
    IF (KEEP(KH) .EQ. 1) GO TO 1140
    LCF = 0
    IF (KH .GT. 17) LCF = IB - NVC
    KPCT1 = 0
    KPCT2 = 0
    KSP = KH
    IF (KH .GT. 17) KSP = 17
    IF (KH .LT. 2) GO TO 820
    DO 810 ICT = 1, KSP
  810   KPCT1 = KPCT1 + 1 - KEEP(ICT)
    GO TO 830
  820   KPCT1 = 1
  830   CONTINUE
    IF (KH .LT. 18) GO TO 850
    DO 840 ICT = 18, KH
  840   KPCT2 = KPCT2 + 1 - KEEP(ICT)
    GO TO 860
  850   KPCT2 = 1
  860   CONTINUE
    II = (KPCT1 + NLC*(KPCT2 - 1) + LCF - 1) * KNOBS(LC1) + IX
  IF (KH .EQ. 1) GO TO 1110
  IF (KH .EQ. 7) GO TO 1120
  IF (KH .EQ. 22) GO TO 1130
  IF (KH .NE. 18) GO TO 870
  IF (MODE .LE. 0) GO TO 870
  IF (IPB .EQ. 1) GO TO 870
  IF (IB .GT. NVC) OBS(II) = (BR(IX) - EL3A(IB)) / HLA(IB)
  GO TO 1140
870   CONTINUE
  IF ((KH - 11)**2 .LE. 9) GO TO 880
  IF ((2*KH - 39)**2 .LE. 9) GO TO 880
  IF ((2*KH - 9)**2 .EQ. 9) GO TO 880
  GO TO 940
880   CONTINUE
  IF (LOW(KH) .EQ. 1) GO TO 910
  M1H = MMSAVH(MM1)
  DO 890 IRE = 1, M1H
    RV(IRE) = SRV(IRE)
    GRX(IRE) = SGRX(IRE)
    GRY(IRE) = SGRY(IRE)
    GRZ(IRE) = SGRZ(IRE)
890   SLUMP1(IRE) = SLMP1(IRE)
  M2H = MMSAVH(MM2)
  DO 900 IRE = 1, M2H
    RVQ(IRE) = SRVQ(IRE)
    GRXQ(IRE) = SGRXQ(IRE)
    GRYQ(IRE) = SGRYQ(IRE)
    GRZQ(IRE) = SGRZQ(IRE)
900   SLUMP2(IRE) = SLMP2(IRE)
  GO TO 940
```

```
910   CONTINUE
       M1L = MMSAVL(MM3)
       DO 920 IRE = 1, M1L
         RV(IRE) = SRVL(IRE)
         GRX(IRE) = SGRXL(IRE)
         GRY(IRE) = SGRYL(IRE)
         GRZ(IRE) = SGRZL(IRE)
920   SLUMP1(IRE) = SLMP1L(IRE)
       M2L = MMSAVL(MM4)
       DO 930 IRE = 1, M2L
         RVQ(IRE) = SRVQL(IRE)
         GRXQ(IRE) = SGRXQL(IRE)
         GRYQ(IRE) = SGRYQL(IRE)
         GRZQ(IRE) = SGRZQL(IRE)
930   SLUMP2(IRE) = SLMP2L(IRE)
940   CONTINUE
       DL = DEL(KH)
       IF (ISYM .EQ. 1) DL = .5 * DEL(KH)
       SIGN = 1.
       ISS = 1
       DO 950 IH = 1, 22
950   DLIF(IH) = 0.
       IF (KS .LE. 2) GO TO 960
       ISS = IS
       DLIF(KH) = 1.
960   CONTINUE
       KFLAG = 0
       COUNTER = 1
       TEMP = 0.0
       DO 1100 IL = 1,ISS
       CALL ITER(E,SIGN,PERR,IL,ELIT,IX,
      1   F1,F2,PSHIFT,XINCL,GR1,GR2,
      2   TAVH,TAVC,ALB1,ALB2,PHSV,PCSV,
      3   RM, HLA(IB),CLA(IB),X1A(IB),X2A(IB),KH,KS,LOW(KH),
      4   RV,GRX,GRY,GRZ,RVQ,GRXQ,GRYQ,
      5   GRZQ,MMSAVH,FR1,FR2,HLD,N1,N2,
      6   MOD,THE,SNTHH,CSTHH,SNFIH,MODE,
      7   SLUMP1,SLUMP2,THETA,RHO,AA,BB,
      8   KF1,KF2,D,WLA(IB),SM1,SM2,TPH,TPC,
      9   SBRH,SBRC,PHAS(IX),HOT,COOL,TLDH,
      +   MMSAVL,N1L,N2L,SNTHL,CSFIL,TLDL,
      +   EL3A(IB),VC1,VC2,BL(IX),OBS(II),
      +   CSTHL,SNFIL,KFLAG,
      +   NMAT,FLUX(IX),DEL,CSFIH,BR(IX),COUNTER,TEMP)
      IF (KFLAG .EQ. 1) THEN
        GO TO 1100
      ELSEIF (KFLAG .EQ. 2) THEN
        II = COUNTER
        OBS(II) = TEMP
        GO TO 1100
      ENDIF
 1100   CONTINUE
      GO TO 1140
C
 1110   IF (IB .LE. NVC) OBS(II) = (BR(IX) - VGA) / A
      GO TO 1140
 1120   IF (IB .LE. NVC) OBS(II) = 1.
      GO TO 1140
 1130   IF (IB .GT. NVC) OBS(II) = 1.
 1140 CONTINUE
```

```
  SIGN = -1.
 1150CONTINUE
 1160 CONTINUE
    GO TO 1240
 1170 JF = 0
    IF (KO .EQ. 0) RETURN
C
    RFLAG = 0
   CALL SCRATCH(OBS,HOLD,KDISK,KEEP,IFDER,IFM,
   1 IFR,KEP,KNOBS,LC1,NCOF,MAT,KTR,
   2 NCOEFF,NOBS,RFLAG)
    IF (RFLAG .EQ. 1) RETURN
C
 1240 WRITE (6,110)
    WRITE (6,100) (KEEP(I),I=1,22), IFDER, IFM, IFR
    NOBS = KNOBS(LC1)
    WRITE (6,120)
    IF (IFDER .EQ. 0) KTR = 5
    WRITE (6,250)
    WRITE (6,120)
    DO 1300 IB = 1, NLVC
IST = KNOBS(IB) + 1
IB1 = IB + 1
ISP = KNOBS(IB1)
DO 1300 I = IST, ISP
 IF (KOSQ .EQ. 1) GO TO 1250
 IF (IBEF .EQ. 1) GO TO 1250
 II = II + 1
 OBS(II) = FLUX(I) -[illegible]B(I)
 1250 GO TO (1260, 1270, 1280, 1290, 1300), KTR
```

```
1260  WRITE (6,10) (OBS(J),J=I,NCOEFF,NOBS)
      GO TO 1300
1270  WRITE (6,20) (OBS(J),J=I,NCOEFF,NOBS)
      GO TO 1300
1280  WRITE (6,50) (OBS(J),J=I,NCOEFF,NOBS)
      GO TO 1300
1290  WRITE (6,60) (OBS(J),J=I,NCOEFF,NOBS)
1300 CONTINUE
1310 CONTINUE
        IF (KO .LE. 1) GO TO 1340
        IF (IBEF .EQ. 1) GO TO 1340
        DO 1320 J = 1, NCOEFF
1320 HOLD(J) = OBS(J)
        IF (KDISK .EQ. 0) GO TO 1330
        OPEN(UNIT=9,FILE='DC.SCRATCH')
        REWIND 9
        WRITE (9,30) (OBS(J),J=1,NCOEFF)
1330 CONTINUE
1340 WRITE (6,120)
        DO 1370 IB = 1, NLVC
IST = KNOBS(IB) + 1
IB1 = IB + 1
ISP = KNOBS(IB1)
NOIS = NOISE(IB)
DO 1370 I = IST, ISP
  IF (IB .GT. NVC) GO TO 1350
  ROOTWT = SQRT(WT(I)) / (100.*SIGMA(IB))
  GO TO 1360
1350 ROOTWT = SQRT(WT(I))/(100.*SIGMA(IB)*SQRT(FLUX(I))**NOIS)
1360  CONTINUE
```

```
 DO 1370 LOB = I, NCOEFF, NOBS
1370 OBS(LOB) = OBS(LOB) * ROOTWT
     IF (IFDER .NE. 0) WRITE (6,260)
     IF (IFDER .NE. 0) WRITE (6,120)
     DO 1420 I = 1, NOBS
GO TO (1380, 1390, 1400, 1410, 1420), KTR
 1380WRITE (6,10) (OBS(J),J=I,NCOEFF,NOBS)
GO TO 1420
 1390WRITE (6,20) (OBS(J),J=I,NCOEFF,NOBS)
GO TO 1420
 1400WRITE (6,50) (OBS(J),J=I,NCOEFF,NOBS)
GO TO 1420
 1410WRITE (6,60) (OBS(J),J=I,NCOEFF,NOBS)
 1420 CONTINUE
 1430 CALL SQUARE(OBS, NOBS, MAT, OUT, PE, D, CN, CNN, S, CCL,
LL, MM)
     MSQ = MAT * MAT
     IF (IFM .EQ. 0) GO TO 1450
     WRITE (6,120)
     WRITE (6,210)
     WRITE (6,120)
     DO 1440 JR = 1, MAT
 1440 WRITE (6,180) (CN(JX),JX=JR,MSQ,MAT), CCL(JR)
     WRITE (6,120)
     WRITE (6,220)
     WRITE (6,120)
 1450 CONTINUE
     NO1 = 15
     NO2 = 16
     NRM = 17
     DO 1460 IRM = 1, 16
```

```
IF (IRM .LE. 15) NO1 = NO1 - KEEP(IRM)
NO2 = NO2 - KEEP(IRM)
 1460 NRM = NRM - KEEP(IRM)
    CORO1 = 1 - KEEP(15)
    CORO2 = 1 - KEEP(16)
    CORQ = 1 - KEEP(17)
    DO 1480 JM = 1, MAT
DO 1470 JQ = 1, MAT
 JT = JM + MAT * (JQ - 1)
 IJM = (MAT + 1) * (JM - 1) + 1
 IJQ = (MAT + 1) * (JQ - 1) + 1
1470V(JQ) = CNN(JT) / DSQRT(CNN(IJM)*CNN(IJQ))
1480 WRITE (6,180) (V(IM),IM=1,MAT)
    CO1Q = V(NO1) * CORQ * CORO1
    CO2Q = V(NO2) * CORQ * CORO2
    IF (IFM .EQ. 0) GO TO 1520
    WRITE (6,120)
    WRITE (6,230)
    WRITE (6,120)
    CALL DGMPRD(CN, CNN, CNOUT, MAT, MAT, MAT)
    DO 1490 J8 = 1, MAT
1490 WRITE (6,180) (CNOUT(J7),J7=J8,MSQ,MAT)
    WRITE (6,120)
    WRITE (6,240)
    WRITE (6,120)
    ANSCH = 0.D0
    DO 1510 J5 = 1, MAT
V(J5) = 0.D0
DO 1500 J6 = 1, MAT
 I9 = J5 + MAT * (J6 - 1)
```

```
1500V(J5) = OUT(J6) * CN(I9) + V(J5)
ERR = V(J5) - CCL(J5)
1510 ANSCH = ANSCH + DABS(ERR)
     WRITE (6,190) (V(J4),J4=1,MAT)
     WRITE (6,120)
     WRITE (6,200) ANSCH
1520 CONTINUE
     WRITE (6,120)
     WRITE (6,120)
     IE = MAT
     IF (MAT .GT. 16) IE = 16
     WRITE (6,310)
     WRITE (6,10) (OUT(I),I=1,IE)
     WRITE (6,120)
     WRITE (6,320)
     WRITE (6,10) (PE(I),I=1,IE)
     IF (MAT .LE. 16) GO TO 1530
     WRITE (6,120)
     WRITE (6,120)
     WRITE (6,310)
     WRITE (6,10) (OUT(I),I=17,MAT)
     WRITE (6,120)
     WRITE (6,320)
     WRITE (6,10) (PE(I),I=17,MAT)
1530 RESSQ = 0.
     JST = MAT * NOBS + 1
     DO 1540 JRES = JST, NCOEFF
1540 RESSQ = RESSQ + OBS(JRES) ** 2
     WRITE (6,120)
     WRITE (6,330)
```

```
      WRITE (6,70) RESSQ, S, D
      IBEF = 1
      IF (IFR .EQ. 0) GO TO 1170
      WRITE (6,110)
      WRITE (6,120)
      WRITE (6,120)
      WRITE (6,510)
      WRITE (6,120)
      WRITE (6,120)
      WRITE (6,540)
      WRITE (6,120)
      DO1 = PE(NO1) * CORO1
      DO2 = PE(NO2) * CORO2
      DQ = PE(NRM) * CORQ
      COQ = CO1Q
      F = F1
      DP = 1. - E
      OME = PHSV
      DOM = DO1
      KOMP = 0
 1550 CONTINUE
      KOMP = KOMP + 1
      DO 1560 KD = 1, 4
TH = XTHA(KD)
FI = XFIA(KD)
CALL ROMQSP(OME, RM, F, DP, E, TH, FI, R, DRDO, DRDQ, DODQ,
   1    KOMP, MODE)
DR   =   SQRT(DRDQ**2*DQ**2   +   DRDO**2*DOM**2   +
2.*COQ*DRDQ*DRDO*DQ*
   1  DOM)
WRITE (6,550) KOMP, ARAD(KD), R, DRDO, DRDQ, DR
```

```
1560 CONTINUE
   DO2DQ = DODQ
   IF (KOMP .EQ. 1) DO1DQ = DODQ
   COQ = CO2Q
   F = F2
   OME = PCSV
   DOM = DO2
   WRITE (6,120)
   IF (KOMP .EQ. 1) GO TO 1550
   WRITE (6,120)
   WRITE (6,520)
  IF (KOMP .EQ. 2) WRITE (6,530) DO1DQ, DO2DQ, CO1Q, CO2Q, DO1,
DO2,
  1DQ
   GO TO 1170
   END
```

A.3. Subroutine ATM : As detailed.

Note : There is a problem with the Kurucz model for stars with Teff between 8000 and 9000 K. (see Wade,R.A. and Rucinski,S.M. : 1983,Astron. Astrophys. Suppl. Ser.,60 ,471).

A.4. Subroutine BB1 : This subroutine makes a preliminary calculation of orbital elements for each phase of the light/radial velocity curves. It then makes a call to MODLOG to determine the geometry of the system.A system which has non-zero eccentricity requires a further calculation of the respective components' volumes at the different phases.

A.5. Subroutine BB2 : This subroutine performs four functions :

i) Calculation of further geometric factors ie. separation and eccentric anomaly for a given phase.

ii) Calls LCR to calculate the contributions to the flux from gravity-darkening, etc.

iii) Calls LIGHT to determine the "visibility" of a component and the total flux observed at that phase.

iv) BB2 will also calculate the radial velocity for this phase if spectroscopic data are being analysed.

A.6. Subroutine BOLO : This subroutine determines the bolometric correction used in the calculation of R, the "reflection effect".

Strategy : The correction for the reflection effect is given by

RATBOL = 10**(.4*DELMB)

where DELMB = DELMV + (BC2 - BC1).

DELMV is a measure of the ratio of the temperature of the two stars before a bolometric correction is made. (BC2 - BC1) is a measure of the difference in energy flux between the two stars due to the differences in the true (bolometric) flux.

A.7. Subroutine CALC1 : This subroutine is used to make a preliminary calculation of our system dimensions (via BB1). If the eccentricity, e is not equal to zero a calculation of periastron potentials (P1AP,P2AP) with the corresponding derivatives is made. This will then be used to calculate the variation in surface area that occurs throughout the binary's cycle (via VOLUME).

```
$LARGE
      SUBROUTINE CALC1(RV,GRX,GRY,GRZ,RVQ,GRXQ,GRYQ,GRZQ,
                       MMSAVH,FR1,FR2,HLD,RM,PHSV,PCSV,
             GR1,GR2,ALB1,ALB2,N1,N2,F1,F2,
MOD,XINCL,THE,SNTHH,CSTHH,SNFIH,CSFIH,MODE)


 COMMON /ECCEN/ EC, A, PERIOD, VGA, SINI, VF, VFAC, VGAM,
     1   VOL1,VOL2,IFC
      COMMON /DPDX/ DPDX1, DPDX2, POT1, POT2
          CALL BB1(RV, GRX, GRY, GRZ, RVQ, GRXQ, GRYQ, GRZQ,
MMSAVH, FR1,
     1 FR2, HLD, RM, PHSV, PCSV, GR1, GR2, ALB1, ALB2, N1, N2, F1,
     2 F2, MOD, XINCL, THE, SNTHH, CSTHH, SNFIH, CSFIH, MODE)
      E = EC
      IF (E .EQ. 0.) GO TO 680
      DAP = 1. + E
      FACT = 1 - E*E
      P1AP = PHSV - 2. * E * RM / FACT
```

```
    CALL VOLUME(VOL1, RM, P1AP, DAP, F1, N1, N1, KFD, 1, RV, GRX,
GRY,
     1   GRZ, RVQ, GRXQ, GRYQ, GRZQ, MMSAVH, FR1, FR2, HLD,
SNTHH,
    2  CSTHH, SNFIH, CSFIH, SUMMD, SMD, 2)
    DPDX1 = (PHSV - P1AP) * FACT * .5 / E
    P2AP = PCSV - 2. * E / FACT
    CALL VOLUME(VOL2, RM, P2AP, DAP, F2, N2, N1, KFD, 2, RV, GRX,
GRY,
     1   GRZ, RVQ, GRXQ, GRYQ, GRZQ, MMSAVH, FR1, FR2, HLD,
SNTHH,
    2  CSTHH, SNFIH, CSFIH, SUMMD, SMD, 2)
    DPDX2 = (PCSV - P2AP) * FACT * .5 / E
  680 CONTINUE
    RETURN
    END
```

A.8. Subroutine DCDRIVE : The driver for the differential corrections portion of the WD code. This subroutine allocates memory for the various parameters.

Note : If one wishes to increase the grid fineness: N1,N2,etc. for a solution you must make sure sufficient memory has been declared. As used in this paper the grid fineness was set to a maximum of N1=N2=30.

```
$LARGE
    DIMENSION PHAS(250), FLUX(250), BR(230), OBS(4000), OUT(037),
    1PE(037), CN(1369), CCL(37), LL(37), MM(37), HLA(15),
    2CLA(15), X1A(15), X2A(15), EL3A(15), WLA(15), NOISE(15),
    3KNOBS(17), RV(0762), GRX(0762), GRY(0762), GRZ(0762),
    4RVQ(0762), GRXQ(0762), GRYQ(0762), GRZQ(0762),
    5SLUMP1(0762), BL(0762), SLUMP2(0762), MMSAVL(100),
    1THETA(260), RHO(260), AA(20), BB(20), SIGMA(15),
    2WT(250), FR1(0762), FR2(0762), HOLD(4000), HLD(4000),
    3CNN(1369), V(60), CNOUT(1369), SNTHH(65), CSTHH(65),
    4SNFIH(1600), CSFIH(1600), SNTHL(65), CSTHL(65),
     5SNFIL(1600), CSFIL(1600), MMSAVH(124), TLDH(1600),
     6TLDL(1600)
C    VERSION OF NOV. 11, 1983. DESTROY EARLIER VERSIONS.
C140
      DOUBLE PRECISION CN, CCL, OUT, CNN, V, CNOUT
      CALL DC(PHAS, FLUX, BR, OBS, OUT, PE, CN, CCL, LL, MM, HLA,
CLA,
     1 X1A, X2A, EL3A, WLA, NOISE, KNOBS, RV, GRX, GRY, GRZ, RVQ,
     2 GRXQ,GRYQ,GRZQ,SLUMP1,BL,SLUMP2,MMSAVL,
     5 THETA, RHO, AA, BB, SIGMA, WT, FR1, FR2, HOLD, HLD, CNN,
     6 SNTHH, CSTHH, SNFIH, CSFIH, SNTHL, CSTHL, SNFIL, CSFIL,
TLDH,
     7 TLDL, V, CNOUT, MMSAVH)
      STOP
      END
```

A.9. Subroutine DGMPRD : This subroutine calculates the product of two matrices defining a system of equations. The result may then be used by other subroutines to minimize a solution.

Strategy : As an example, upon being called by NEKMIN one has the following input parameters :

A(1) = DN(1) = OMX ; A(2) = DN(2) = OMZ

A(3) = DN(3) = OMXX ; A(4) = DN(4) = OMZX

EN(1) = DELOM ; EN(2) = -DN(1) = -OMX

Our system then is given as :

DN(1) DN(3)EN(1)   DN(1)EN(1) + DN(3)EN(2)

=

DN(2) DN(4)EN(2)   DN(2)EN(1) + DN(4)EN(2)

OMX*DELOM - OMX*OMXX   R(1)   OUT(1)

=   =   =

OMZ*DELOM - OMX*OMZX   R(2)   OUT(2)

In this case, the output OUT(1), OUT(2) serves as the correction for an iterative process (The least-squares fitting of data).

A.10. Subroutine DMINV : This subroutine is used in conjunction with DGMPRD to solve a system of equations. DMINV performs a LU decomposition necessary for Gaussian elimination with back substitution.

A.11. Subroutine DURA : This program calculates the radius and potential associated with a x-ray eclipse i.e. it overrides any input potential to guarantee agreement with observation.

A.12. Subroutine ELLONE : Based upon the Roch model this subroutine determines the potentials and point radii associated with the Roche lobes for a binary system.

A.13. Subroutine FOUR : This subroutine calculates the fourier representation for the projected boundary image of the system. LC and DC will then use this in the calculation of eclipse events (total luminosity, duration,etc.).

A.14. Subroutine INOUT1 : A simple input/output subroutine.

```
$LARGE
    SUBROUTINE INOUT1(NS,NVC,NLVC,PHAS,FLUX,
   1WT,NOBS,KNOBS,LC1)
    DIMENSION PHAS(250),FLUX(250),WT(250),KNOBS(17)
    INTEGER NY,I1,I2,I3,I4,I5,NI,NOBS,NS,LCV
    NY = 0
    NS = 1
    KNOBS(1) = 0
    OPEN(UNIT=6,FILE='DC.OUT')
    WRITE(6,270)
    OPEN(UNIT=5,FILE='WIL3.DAT')
    DO 780 LCV = 1, NLVC
WRITE(6,120)
IF (LCV .LE. NVC) WRITE(6,280)
```

```
 IF (LCV .GT. NVC) WRITE(6,290)
 DO 760 I = NS, 7000
  I1 = 5 * (I - 1) + NY + 1
  I2 = I1 + 1
  I3 = I2 + 1
  I4 = I3 + 1
  I5 = I4 + 1
  READ (5,140) PHAS(I1), FLUX(I1), WT(I1), PHAS(I2),
    1  FLUX(I2),
    2  WT(I2), PHAS(I3), FLUX(I3), WT(I3), PHAS(I4), FLUX(I4),
    3  WT(I4), PHAS(I5), FLUX(I5), WT(I5)
  WRITE(6,150) PHAS(I1), FLUX(I1), WT(I1), PHAS(I2),
    1      FLUX(I2), WT(I2), PHAS(I3), FLUX(I3),
    2      WT(I3), PHAS(I4), FLUX(I4), WT(I4),
    3      PHAS(I5), FLUX(I5), WT(I5)
  IF (PHAS(I1) .LT. 10.) GO TO 760
  NI = PHAS(I1) - 10.
  NY = NY + NI
  NOBS = 5 * (I - NS - 1) + NI
  GO TO 770
  760CONTINUE
  770NS = I - 1
 LC1 = LCV + 1
  780 KNOBS(LC1) = NOBS + KNOBS(LCV)
 C
  120 FORMAT (' ')
  140 FORMAT (5(F6.4,F7.4,F3.1))
  150 FORMAT (1X, 5(F7.4,F7.4,F4.1))
  270 FORMAT (33X,'OBSERVATIONS')
  280 FORMAT ('  PHASE  V RAD  WT  PHASE  V RAD  WT  PHASE  V
 RAD  WT  PHASE  V RAD  WT  PHASE
```

```
   1  V RAD  WT')
 290 FORMAT (' PHASE  LIGHT  WT  PHASE  LIGHT  WT  PHASE  LIGHT
WT  PHASE  LIGHT  WT  PHASE
   1  LIGHT  WT')
    RETURN
    END
```

A.15. Subroutine INPUT1 : A simple input subroutine.

```
$LARGE
    SUBROUTINE INPUT1(DEL1,KEP1,IFDER,IFM,IFR,KO,KDISK,
   1ISYM,MODE,N1,N2,
   2N1L,N2L,THE,E,F1,F2,
   3PSHIFT,XINCL,GR1,GR2,TAVH,TAVC,
   4ALB1,ALB2,PHSV,PCSV,RM)
    DIMENSION DEL1(22),KEP1(22)
     COMMON /ECCEN/ EC, A, PERIOD, VGA, SINI, VF, VFAC, VGAM,
VOL1,
   1    VOL2, IFC
    COMMON /FLVAR/ PER, PSH, DP, DSDUM, EF, EFC, ECOS, PERT,
PHPER,
    1     PCONJ, PHPERI, VSUM1, VSUM2, VRA1, VRA2, VKM1, VKM2,
VUNIT,
   2    VFVU
    COMMON /INVAR/ KH, IPB
    COMMON /INCON/ IFVC1, IFVC2, NLC
    COMMON /NSPT/ NSP1, NSP2, IFAT1, IFAT2
C
    OPEN(UNIT=5,FILE='WIL3.DAT')
    READ (5,90) (DEL1(I),I=2,6),(DEL1(I),I=8,12)
    READ (5,90) (DEL1(I),I=13,21)
    READ (5,100) (KEP1(I),I=1,22), IFDER, IFM, IFR
    READ (5,460) IFVC1, IFVC2, NLC, KO, KDISK, ISYM
   READ (5,430) MODE, IPB, IFAT1, IFAT2, N1, N2, N1L, N2L, PERIOD,
  1THE, VUNIT
   READ (5,440) E, PER, A, F1, F2, VGA, PSHIFT, XINCL, GR1, GR2
   READ (5,450) TAVH, TAVC, ALB1, ALB2, PHSV, PCSV, RM
 90 FORMAT (10(1X,E7.1))
100 FORMAT (1X, 22I1, 1X, I1, 1X, I1, 1X, I1)
430 FORMAT (I2, 1X, I1, 1X, I1, 1X, I1, 1X, 4(I2,1X), E9.4, 1X, F5.4,
   1     1X, F8.3)
440 FORMAT (F5.4, 1X, F6.2, 1X, F7.3, 1X, 2(F6.3,1X), F7.4, 1X, F5.4,
   1     1X, F6.3, 2(1X,F6.3))
460 FORMAT (I1, 1X, I1, 1X, I2, 3(1X,I1))
450 FORMAT (2(F6.4,1X), 2(F6.3,1X), 2(F7.4,1X), F8.5)
   RETURN
   END
```

A.16. Subroutine INPUT2 : A simple input subroutine for some of the control parameters. The polar coordinates for any spots are calculated at this point.

```
$LARGE
                                                   SUBROUTINE
INPUT2(NVC,NVCP,NLVC,NSP1,NSP2,WLA,HLA,CLA,
   1X1A,X2A,SIGMA,EL3A,NOISE,XLAT,XLONG)
    DIMENSION WLA(1),CLA(1),X1A(1),X2A(1),SIGMA(1),
   1NOISE(1),EL3A(1),HLA(1)
    DIMENSION XLAT(2,100),XLONG(2,100)
    COMMON /SPOTS/ SINLAT(2,100), COSLAT(2,100), SINLNG(2,100),
   1    COSLNG(2,100), RADSP(2,100), TEMSP(2,100)
C
    DTR = 1.745329E-2
    IF (NVC .EQ. 0) GO TO 630
   DO 620 I = 1, NVC
   OPEN(UNIT=5,FILE='WIL3.DAT')
 620 READ (5,170) WLA(I),HLA(I),CLA(I),
   1  X1A(I),X2A(I),SIGMA(I)
 630 CONTINUE
   IF (NLVC .EQ. NVC) GO TO 650
   DO 640 I = NVCP, NLVC
 640 READ (5,160) WLA(I), HLA(I),CLA(I),X1A(I),
   1  X2A(I),EL3A(I),NOISE(I),SIGMA(I)
 650 CONTINUE
   NSP1 = 0
   NSP2 = 0
   DO 670 KP = 1, 2
DO 660 I = 1, 100
 READ (5,570) XLAT(KP,I), XLONG(KP,I), RADSP(KP,I),
   1     TEMSP(KP,I)
 IF (XLAT(KP,I) .GE. 200.) GO TO 670
 SINLAT(KP,I) = SIN(DTR*XLAT(KP,I))
 COSLAT(KP,I) = COS(DTR*XLAT(KP,I))
 SINLNG(KP,I) = SIN(DTR*XLONG(KP,I))
 COSLNG(KP,I) = COS(DTR*XLONG(KP,I))
 IF (KP .EQ. 1) NSP1 = NSP1 + 1
 660IF (KP .EQ. 2) NSP2 = NSP2 + 1
 670 CONTINUE
 160 FORMAT (1X, F6.4, 2(1X,F7.4), 2(1X,F5.3), 1X, F6.4, 1X, I1, 1X,
   1    F6.4)
 170 FORMAT (1X, F6.4, 2(1X,F7.4), 2(1X,F5.3), 1X, F6.4)
 570 FORMAT (3F7.2, 2F8.3)
   RETURN
   END
```

A.17. Subroutine ITER : This subroutine performs several functions depending on the iteration. On the first two passes (KS=1,2) it calculates the contribution to the flux for the low precision (low grid fineness) or high precision (at present these include e,w,F1,F2,Po,i,OM1,OM2 and q). After that ITER is only called by DC when a parameter is to be varied. Thus, it forms the necessary residual

TEMP = FLUX(IX) - BR
= O - C

and the necessary derivatives

OBS = (XNR - XLR)/DEL(KH)

where XNR = f(x + delx) or f(x + delx/2)
XLR = f(x) or f(x - delx/2).

This subroutine also calculates the variation in the radial velocity data by setting

ELIT = 0
VC1 = 1, VC2 = 0 first data set
VC1 = 0, VC2 = 1 second data set

```
$LARGE
      SUBROUTINE ITER(E,SIGN,PERR,IL,ELIT,IX,
     1     F1,F2,PSHIFT,XINCL,GR1,GR2,
     2     TAVH,TAVC,ALB1,ALB2,PHSV,PCSV,
     3     RM,HLA,CLA,X1A,X2A,KH,KS,LOW,
     4     RV,GRX,GRY,GRZ,RVQ,GRXQ,GRYQ,
     5     GRZQ,MMSAVH,FR1,FR2,HLD,N1,N2,
     6     MOD,THE,SNTHH,CSTHH,SNFIH,MODE,
     7     SLUMP1,SLUMP2,THETA,RHO,AA,BB,
     8     KF1,KF2,D,WLA,SM1,SM2,TPH,TPC,
     9     SBRH,SBRC,PHAS,HOT,COOL,TLDH,
     +     MMSAVL,N1L,N2L,SNTHL,CSFIL,TLDL,
     +     EL3A,VC1,VC2,BL,OBS,
     +     CSTHL,SNFIL,KFLAG,NMAT,
     +             FLUX,DEL,CSFIH,BR,COUNTER,TEMP)
```

```
C
      DIMENSION DEL(22),FLUX(1),
     1 RV(1),GRX(1),GRY(1),GRZ(1),
     2 RVQ(1),GRXQ(1),GRYQ(1),GRZQ(1),MMSAVH(1),
     3 MMSAVL(1),HLD(1),CSTHL(1),SNFIL(1),
     6TLDL(1),TLDH(1),SNTHH(1),CSTHH(1),SNFIH(1),
     7CSFIH(1),OUT(1),AA(1),
     +BB(1),FR1(1),FR2(1),THETA(1),RHO(1),
     +SLUMP1(1),SLUMP2(1)
      INTEGER COUNTER
      REAL OBS,TEMP
      COMMON /ECCEN/ EC, A, PERIOD, VGA, SINI,
     1   VF, VFAC, VGAM, VOL1,
     2   VOL2, IFC
      COMMON /DPDX/ DPDX1, DPDX2, POT1, POT2
      COMMON /FLVAR/ PER, PSH, DP, DSDUM, EF,
     1   EFC, ECOS, PERT, PHPER,
     2   PCONJ, PHPERI, VSUM1, VSUM2, VRA1,
     3   VRA2, VKM1, VKM2, VUNIT,VFVU
      COMMON /NSPT/ NSP1, NSP2, IFAT1, IFAT2
      COMMON /CNTRL/ DL,MM1,MM2,MM3,MM4,DLIF(22)
      COMMON /VCOM/ SRV(0762),SGRX(0762),SGRY(0762),
     1  SGRZ(0762),SLMP1(0762)
      COMMON /VCOMQ/ SRVQ(0762),SGRXQ(0762),SGRYQ(0762),
     1  SGRZQ(0762),SLMP2(0762)
      COMMON /VCOML/ SRVL(0762),SGRXL(0762),SGRYL(0762),
     1  SGRZL(0762),SLMP1L(0762)
      COMMON /VCOMQL/ SRVQL(0762),SGRXQL(0762),SGRYQL(0762),
     1  SGRZQL(0762),SLMP2L(0762)
      DELS = DL * SIGN
      SIGN = -1.
      EC = E + DELS * DLIF(2)
      PERT = PERR + DELS * DLIF(3)
      FF1 = F1 + DELS * DLIF(4)
      FF2 = F2 + DELS * DLIF(5)
      PSH = PSHIFT + DELS * DLIF(6)
      XINC = XINCL + DELS * DLIF(8)
      G1 = GR1 + DELS * DLIF(9)
      G2 = GR2 + DELS * DLIF(10)
      T1 = TAVH + DELS * DLIF(11)
      T2 = TAVC + DELS * DLIF(12)
      A1 = ALB1 + DELS * DLIF(13)
      A2 = ALB2 + DELS * DLIF(14)
      POT1 = PHSV + DELS * DLIF(15)
      POT2 = PCSV + DELS * DLIF(16)
      RMASS = RM + DELS * DLIF(17)
      HL = HLA + DELS * DLIF(18)
      CL = CLA + DELS * DLIF(19)
      X1 = X1A + DELS * DLIF(20)
      X2 = X2A + DELS * DLIF(21)
      IF (KS .EQ. 1) GO TO 980
      IF (KS .EQ. 2) GO TO 970
      IF (LOW .EQ. 1) GO TO 980
  970      CALL BB1(RV, GRX, GRY, GRZ, RVQ, GRXQ, GRYQ, GRZQ,
MMSAVH,
     1  FR1, FR2, HLD, RMASS, POT1, POT2, G1, G2, A1, A2, N1,
     2  N2, FF1, FF2, MOD, XINC, THE, SNTHH, CSTHH, SNFIH,
     3  CSFIH, MODE)
          CALL BB2(RV, GRX, GRY, GRZ, RVQ, GRXQ, GRYQ, GRZQ,
MMSAVH,
     1  FR1, FR2, HLD, SLUMP1, SLUMP2, THETA, RHO, AA, BB,
      2 POT1, POT2, N1, N2, KF1, KF2, FF1, FF2, D, HL, CL,
      3 X1,X2,G1,G2,WLA,SM1,SM2,TPH,TPC,SBRH,
      4 SBRC,IFAT1,IFAT2,T1,T2,A1,A2,PHAS,RMASS,
      5 XINC, HOT, COOL, SNTHH, CSTHH, SNFIH, CSFIH, TLDH,
      6 MODE)
       GO TO 990
  980     CONTINUE
          CALL BB1(RV, GRX, GRY, GRZ, RVQ, GRXQ, GRYQ, GRZQ,
MMSAVL,
      1 FR1, FR2, HLD, RMASS, POT1, POT2, G1, G2, A1, A2,
      2 N1L, N2L, FF1, FF2, MOD, XINC, THE, SNTHL, CSTHL,
      3 SNFIL, CSFIL, MODE)
          CALL BB2(RV, GRX, GRY, GRZ, RVQ, GRXQ, GRYQ, GRZQ,
MMSAVL,
      1 FR1, FR2, HLD, SLUMP1, SLUMP2, THETA, RHO, AA, BB,
      2 POT1, POT2, N1L, N2L, KF1, KF2, FF1, FF2, D, HL, CL,
      3 X1,X2,G1,G2,WLA,SM1,SM2,TPH,TPC,SBRH,
      4 SBRC,IFAT1,IFAT2,T1,T2,A1,A2,PHAS,RMASS,
      5 XINC, HOT, COOL, SNTHL, CSTHL, SNFIL, CSFIL, TLDL,
      6 MODE)
  990     CONTINUE
       IF (KS .NE. 2) GO TO 1020
       M1H = MMSAVH(MM1)
       DO 1000 IHH = 1, M1H
SRV(IHH) = RV(IHH)
SGRX(IHH) = GRX(IHH)
SGRY(IHH) = GRY(IHH)
SGRZ(IHH) = GRZ(IHH)
SLMP1(IHH) = SLUMP1(IHH)
 1000     CONTINUE
       M2H = MMSAVH(MM2)
```

```
    DO 1010 IHH = 1, M2H
SRVQ(IHH) = RVQ(IHH)
SGRXQ(IHH) = GRXQ(IHH)
SGRYQ(IHH) = GRYQ(IHH)
SGRZQ(IHH) = GRZQ(IHH)
SLMP2(IHH) = SLUMP2(IHH)
 1010     CONTINUE
 1020     CONTINUE
    IF (KS .NE. 1) GO TO 1050
    M1L = MMSAVL(MM3)
    DO 1030 IHL = 1, M1L
SRVL(IHL) = RV(IHL)
SGRXL(IHL) = GRX(IHL)
SGRYL(IHL) = GRY(IHL)
SGRZL(IHL) = GRZ(IHL)
SLMP1L(IHL) = SLUMP1(IHL)
 1030     CONTINUE
    M2L = MMSAVL(MM4)
    DO 1040 IHL = 1, M2L
SRVQL(IHL) = RVQ(IHL)
SGRXQL(IHL) = GRXQ(IHL)
SGRYQL(IHL) = GRYQ(IHL)
SGRZQL(IHL) = GRZQ(IHL)
SLMP2L(IHL) = SLUMP2(IHL)
 1040     CONTINUE
 1050     CONTINUE
    HTT = HOT
    IF (MODE .EQ. - 1) HTT = 0.
    XR = (HTT + COOL + EL3A) * ELIT + VKM1 * VC1
    1     + VKM2 * VC2
```

```
      IF (KS .NE. 1) GO TO 1060
      BL = XR
      KFLAG = 1
      GO TO 1140
 1060     CONTINUE
      IF (KS .NE. 2) GO TO 1070
      BR = XR
      COUNTER = NMAT + IX
      TEMP = FLUX(IX) - XR
      KFLAG = 2
      GO TO 1140
 1070     CONTINUE
      XLR = BR
      IF (LOW .EQ. 1) XLR = BL
      IF (IL .NE. 2) GO TO 1080
      XLR = XR
      GO TO 1090
 1080     XNR = XR
 1090     OBS = (XNR - XLR) / DEL(KH)
 1140CONTINUE
    RETURN
    END
```

A.18. Subroutine KEPLER : This subroutine solves Kepler's equation to determine the true anomaly for the system (the angle past periastron in the orbital plane).

Strategy : Using the relationship

tan(v/2) = sqrt ((1+e)/(1-e)) tan(E/2)   (A.1)

one may find v given E and e. These in turn are found by minimizing the mean anomaly in Kepler's equation

M = E - esinE   (A.2)

via delE = delM*(diE/diM)   (A.3)

to obtain a final E for a given eccentricity, e.

A.19. Subroutine LCDRIVE : The driver for the synthetic light curve program LC, this subroutine allocates memory for the various parameters.

Note : See comment in DCDRIVE.

```
$LARGE
        DIMENSION RV(0762), GRX(0762), GRY(0762), GRZ(0762),
RVQ(0762),
   1GRXQ(0762), GRYQ(0762), GRZQ(0762), SLUMP1(0762),
   2SLUMP2(0762), MMSAVE(124), THETA(260), RHO(260), AA(7),
   3BB(7), FR1(0762), FR2(0762), HLD(0762), SNTH(65),
   4CSTH(65), SNFI(1600), CSFI(1600), TLD(1600)
C    VERSION OF 4/27/83. DESTROY EARLIER VERSIONS.
   CALL LC(RV, GRX, GRY, GRZ, RVQ, GRXQ, GRYQ, GRZQ, SLUMP1,
SLUMP2,
     1    MMSAVE, THETA, RHO, AA, BB, FR1, FR2, HLD, SNTH, CSTH,
SNFI,
   2  CSFI, TLD)
    STOP
    END
```

A.20. Subroutine LCR : For non-zero eccentric orbits (IFC =.1) this subroutine calculates the temperature for each element normalized to the pole ie. it considers gravity-darkening only. This in turn is used by a Planckian function to obtain the surface brightness. A call to BOLO followed by LUMP results in this flux being corrected for reflection.

A.21. Subroutine LIGHT : This subroutine performs the task of determining whether an element on the surface of a star is "visible". To do this it follows the following strategy :

i) determine which star is nearer to an observer,

ii) for the nearer star compare each element with the outline of the system as projected onto the sky (performed by RING). If visible then add the contribution to luminosity/radial velocity. And

iii) repeat for the more distant star.

A.22. Subroutine LUM : The purpose of this subroutine is twofold. First and foremost is the calculation of the flux contribution from each element. This will take into consideration its physical size and local surface gravity. Secondly, calculate the volume of the stars for both cases of zero and non-zero eccentricity.

A.23. Subroutine LUMP : This subroutine calculates the output luminosities for the two components. It performs the correction for the "reflection effect" and determines the contributions from all elements as if they were visible (LIGHT later reanalyzes the geometry to determine if any are hidden for that particular phase).

A.24. Subroutine MATFOR : This subroutine calculates various parameters required in the matrix solution of a least-squares problem ie. matrix size, number of independent variables and number of coefficients.

```
$LARGE
      SUBROUTINE MATFOR(KEEP,MAT,KTR,NCOEFF,
     1KNOBS)
      DIMENSION KEEP(23)
      INTEGER NCOEFF
      COMMON /INCON/  IFVC1,IFVC2,NLC
C
      MATRIX = 18 - KEEP(1) - KEEP(2) - KEEP(3) - KEEP(4)
     1    - KEEP(5) -
     2    KEEP(6) - KEEP(7) - KEEP(8) - KEEP(9)
     3    - KEEP(10) - KEEP(11) -
     4    KEEP(12) - KEEP(13) - KEEP(14)
     5    - KEEP(15) - KEEP(16) - KEEP(17) +
     6    NLC * (5 - KEEP(18) - KEEP(19)
     7    - KEEP(20) - KEEP(21) - KEEP(22))
      MAT = MATRIX - 1
      EM = MATRIX - 15
      KTR = .24 * EM + 2.2
      IF (EM .LE. 1.5) KTR = 1
IF (EM .GT. 12.) KTR = 5
NCOEFF = MATRIX * KNOBS
RETURN
END
```

A.25. Subroutine MODLOG : As the name suggests, the Model Lobe Geometry subroutine determines the "point radii" and potentials for the binary system. Actual calculations of the above quantities are made in ELLONE as determined by the MODE in MODLOG.

Strategy : Given the MODE this subroutine will determine the type of Roche lobe geometry :

i) MODE = -1. This assumes a detached system so we have GR1, GR2, ALB1, ALB2, POTC, POTH set as input. As an added constraint MODLOG calculates OM2 such that it is in agreement with the inclination XINCL, and mass ratio, q.

ii) MODE 0 : As above but without the added constraint.

iii) MODE 1 : For contact systems with e = 0. Regardless of input parameters MODLOG selects g2 = g1, A2 = A1, OM2 = OM1, X2 = X1 and T2 = T1 (a2/a1)**.25g1 (polar temp.) ie. our secondary input parameters are ignored.

iv) MODE 2 : Similar to MODE 0 but L2 is coupled to T2 through the Planckian function.

v) MODE 3 : Similar to MODE 1 i.e. contact but T2 may be adjusted.

vi) MODE 4 : MD4 = 1 and ELLONE is called with the condition that the primary fills its Roche lobe.

v) MODE 5 : As above but MD5 = 1 and the secondary fills its Roche lobe.

vii) MODE 6 : ELLONE is called twice here to determine the potentials for filled Roche lobes of both stars ie. a double-contact system.

The second part of this subroutine is only used for X-ray eclipses to confirm that the supplied duration of the X-ray eclipse agrees with the geometry of the lobes.

Note : The scaling of POTC within this subroutine (and throughout the WD code)

POTC = q*POTC + .5(1 - q)

is to account for the change in calculations if the coordinate system is changed to the secondary star.

A.26. Subroutine NEKMIN : As the name suggests this subroutine calculates the location of the minimum neck width for a contact system. With this information it may be determined whether one or both stars overflow their respective critical Roche potentials. This surface does not necessarily coincide with the inner Lagrangian point. L1.

Strategy : As a two step process the algorithm is :

i) given the input potential, one minimizes x. Then

ii) starting with the above x-coordinate one will further minimize z.

A.27. Subroutine RING : This subroutine examines the complex geometry associated with contact systems. In particular it determines whether a grid element is within the "neck" of the common envelope. A fully enclosed element will not make a contribution to the observed flux; a fully exposed element will provide the full calculated flux and a partially exposed element some fraction thereof. (See Biermann for a more detailed discussion.)

A.28. Subroutine ROMQSP : This subroutine makes calculations for the radii and relevant derivatives of the radii for the pole, point, side and back of the stars. The importance of this output is twofold.

i) One may take these radii and use them in generating a computer image of the system.

ii) One may estimate the corrections (del OM's) from the output derivatives diOM/dir's. From the output value of diOM/dir and with the assumption that dir should be approximately 1% of r this allows one to check the size of delOM.

A.29. Subroutine SCRATCH : This subroutine generates the scratch pad (DC.SCRATCH) where the derivatives and residuals are stored. It is from this location that the subsequent runs of subsets may be run.

```
$LARGE
    SUBROUTINE SCRATCH(OBS,HOLD,KDISK,KEEP,IFDER,IFM,
   1 IFR,KEP,KNOBS,LC1,NCOF,MAT,KTR,
   2 NCOEFF,NOBS,RFLAG)
                                                   DIMENSION
NSHIFT(23),OBS(1),KEEP(23),KEP(22),HOLD(1),KNOBS(1)
    COMMON /INCON/ IFVC1,IFVC2,NLC
C
    DO 1180 J = 1, NCOF
```

```
 1180 OBS(J) = HOLD(J)
    IF (KDISK .EQ. 0) GO TO 1190
    OPEN(UNIT=5,FILE='WIL3.DAT')
    OPEN(UNIT=9,FILE='DC.SCRATCH')
    REWIND 9
    READ (9,30) (OBS(J),J=1,NCOF)
 1190 READ (5,100) (KEEP(I),I=1,22), IFDER, IFM, IFR
    IF (KEEP(1) .EQ. 2) THEN
 RFLAG = 1
 RETURN
    ENDIF
    DO 1200 I = 1, 22
 1200 IF (KEP(I) .EQ. 1) KEEP(I) = 1
C
     NOBS = KNOBS(LC1)
    CALL MATFOR(KEEP,MAT,KTR,NCOEFF,
    1 KNOBS(LC1))
C
    KC = 1
    NSHIFT(1) = 0
    DO 1210 I = 2, 23
IF (I .GT. 18) KC = NLC
KE = 0
J = I - 1
IF (KEEP(J) .GT. KEP(J)) KE = 1
 1210 NSHIFT(I) = NOBS * KE * KC + NSHIFT(J)
    NOBBS = NOBS
    JF = 0
    DO 1230 I = 1, 23
IF (KEEP(I) .EQ. 1) GO TO 1230
IF (I .GT. 17) NOBBS = NOBS * NLC
IF (I .EQ. 23) NOBBS = NOBS
DO 1220 J = 1, NOBBS
 JF = JF + 1
 KX = JF + NSHIFT(I)
1220OBS(JF) = OBS(KX)
1230 CONTINUE
 30 FORMAT (20A4)
 100 FORMAT (1X, 22I1, 1X, I1, 1X, I1, 1X, I1)
    RETURN
    END
```

A.30. Subroutine SINCOS : The purpose of this subroutine is to calculate the colatitude and longitude of the various surface elements of the two stars.The contribution in flux to a light curve appears as the sum of discrete sources centred in these irregularly shaped polygons.

Note : Distorted stars require a fine grid to ensure good representation of the equatorial regions. In general, the ratio of grid fineness should vary as the square of the radius of the star.

A.31. Subroutine SORTER : This subroutine performs a sort of angles within an array into ascending order.

A.32. Subroutine SPOT : This subroutine makes a simple calculation for the increase or decrease of local temperature due to the presence of a spot. This is the local temper/ature found from consideration of local gravity only.

Note: This may be used to simulate the affects of intervening stellar material/streams on the light variation.

A.33. Subroutine SQUARE : This subroutine solves the system of NE's describing the light/radial velocity variation of the system. As such it makes use of Gaussian elimination with pivoting.

A.34. Subroutine SURFAS : SURFAS performs three basic functions. For a given potential it :

i) calculates the radius of a given star in all directions,

ii) calculates the rectangular components of the potentials at the surface of a given star and,

iii) checks with RING to make sure all surface components do not fall into the common envelope if it is a contact system.

A.35. Subroutine VOLUME : This subroutine calculates the volumes of the two components. Normally, for zero eccentricity orbits IFC=1. For non-zero eccentricity orbits this subroutine will calculate the volume as a function of phase.

APPENDIX B

Original Programs

This appendix is a list of original programs used in the data reduction process. All programs were executed on the AMDAHL mainframe computer at the University of Alberta.

B.1. PROGRAM PHASER

This program converts output from the current data reduction program (HISP) to produce a phased, average magnitude.

Compilation of this program follows from:

```
 $RUN *FORTRANVS SCARDS=PHASER SPUNCH=PHASER.BJ T=1
```

Control parameters for determination of phase ie. reference Julian date and period as well as the averaging interval is provided in the file PHASE.CON

Execution of the program follows as

```
$RUN PHASER.OBJ 6=PHASE.CON 5=INDATA 7=OUTDATA T=1
      DIMENSION MAG(5000),HJDATE(5000),PHASE(5000),
   * ADDP(5000),ADDM(5000)
   * INTEGER UTHR,UTHM,HR,HM,COUNT,COUNT2,AVEINT,
   * UPPER,LOWER,COUNT3
     REAL*8 UTHS,HS,VAR,VSKY,VMVSKY,COMP,SMOOTH,CSKY,
   * CMCSKY,AMASS,ADJCNT,MAG,HJDATE,PADD,MAGADD,
   * PERIOD,REF,ADDP,ADDM
     COMMON/STORE1/PERIOD,REF/STORE2/MAG/
   * STORE3/PHASE/STORE4/ADDP/STORE5/ADDM
   * /STORE6/COUNT3,AVEINT,COUNT
   * /STORE7/HJDATE
C
    READ(6,*) PERIOD,REF,AVEINT
    COUNT=1
    DO 10 I=1,99999,1
     IF(COUNT.EQ.5001) THEN
     WRITE(7,*)'TOO MANY DATA POINTS'
```

```
  STOP
 ELSE
  READ(5,1,END=11) UTHR,UTHM,UTHS,HR,HM,HS,
 *  VAR,VSKY,VMVSKY,SMOOTH,
 *  COMP,CSKY,CMCSKY,AMASS,ADJCNT,
 *  MAG(I),HJDATE(I)
 ENDIF
 COUNT=COUNT+1
10 CONTINUE
11 CONTINUE
  COUNT=COUNT-1
  CALL AVE
  DO 30 J=1,(COUNT3-1),1
   WRITE(7,2)ADDP(J),ADDM(J)
30 CONTINUE
1  FORMAT(1X,2I3,F5.1,1X,2I3,F5.1,F9.0,F6.0,F9.0,F8.0,F9.0,
 * F7.0,F9.0,F8.3,F9.0,F7.3,F14.4)
2  FORMAT(1X,F5.4,2X,F6.4)
   STOP
   END
```

B.1.1. FUNCTION PDATA

A simple function to calculate the phase of each Julian date relative to the provided reference date.

```
    FUNCTION PDATA(DATE)
    REAL*8 DATE,DIFF,FPER,FREF
    COMMON /STORE1/FPER,FREF
C
 DIFF=DATE-FREF
     DIFF=DIFF/FPER
     PDATA=DIFF-AINT(DIFF)
```

```
C
      RETURN
      END
```

B.1.2. SUBROUTINE AVE

This subroutine simply calculates the average phase and magnitude for a given interval. A provision is made in the algorithm for the case of the remaining data set being smaller than the specified interval ie. the average over the remaining data set is calculated.

```
      SUBROUTINE AVE
      DIMENSION MAG1(5000),PHASE1(5000),ADP1(5000),
     * ADM1(5000),DATE(5000)
      REAL*8 MADD1,PADD1,MAG1,PHASE1,ADP1,ADM1,DATE
     *,XPER,XREF
      INTEGER UPPER,UPPER2,LOWER,CNT3,INAVE,DIV1,
     * REM,CNT1
      COMMON/STORE1/XPER,XREF/STORE2/MAG1
     * /STORE3/PHASE1/STORE4/ADP1/STORE5/ADM1
     * /STORE6/CNT3,INAVE,CNT1/STORE7/DATE
C
      DO 20 J=1,CNT1,1
       PHASE1(J)=PDATA(DATE(J))
 20   CONTINUE
      MADD1=0.0
      PADD1=0.0
      LOWER=1
      UPPER2=INAVE
      UPPER=CNT1
      CNT3=1
      IF (CNT1.LE.INAVE) THEN
       DO 50 J=1,CNT1,1
        MADD1=MADD1+MAG1(J)
      PADD1=PADD1+PHASE1(J)
50    CONTINUE
     ADM1(CNT3)=MADD1/CNT1
     ADP1(CNT3)=PADD1/CNT1
     CNT3=CNT3+1
     GO TO 110
    ELSE
     DIV1=INT(CNT1/INAVE)
     UPPER=INAVE*DIV1
     REM=CNT1-UPPER
     LOWER=1
     UPPER2=INAVE
    IF(REM.NE.0)THEN
     DO 60 J=1,DIV1,1
      DO 70 I=LOWER,UPPER2,1
       MADD1=MADD1+MAG1(I)
       PADD1=PADD1+PHASE1(I)
70   CONTINUE
     ADM1(CNT3)=MADD1/INAVE
     ADP1(CNT3)=PADD1/INAVE
     CNT3=CNT3+1
     LOWER=LOWER+INAVE
     UPPER2=UPPER2+INAVE
     MADD1=0.0
     PADD1=0.0
60 CONTINUE
    IF(REM.EQ.1)THEN
     ADM1(CNT3)=MAG1(CNT1)
     ADP1(CNT3)=PHASE1(CNT1)
     CNT3=CNT3+1
```

```
       GO TO 110
     ELSE
       DO 80 I=(UPPER+1),CNT1,1
         MADD1=MADD1+MAG1(I)
         PADD1=PADD1+PHASE1(I)
80  CONTINUE
       ADM1(CNT3)=MADD1/REM
       ADP1(CNT3)=PADD1/REM
       CNT3=CNT3+1
       GO TO 110
      ENDIF
      ELSE
       DO 90 J=1,DIV1,1
         DO 100 I=LOWER,UPPER2,1
           MADD1=MADD1+MAG1(I)
           PADD1=PADD1+PHASE1(I)
100   CONTINUE
       ADM1(CNT3)=MADD1/INAVE
       ADP1(CNT3)=PADD1/INAVE
        CNT3=CNT3+1
       LOWER=LOWER+INAVE
       UPPER2=UPPER2+INAVE
       MADD1=0.0
       PADD1=0.0
90      CONTINUE
       GO TO 110
     ENDIF
    ENDIF
C 110   RETURN
    END
```

B.2. PROGRAM GAINCORR

This program calculates the gain correction to account for the difference in the sensitivity of the two channels. Gain corrections are made via a set of piecewise cubic splines (cf. Johnston, p.114-119).

Compilation of this program is achieved by

```
$RUN *PASCALVS SCARDS=GAINCORR SPUNCH=GAIN.OBJ T=1
```

Followed by

```
$RUN GAIN.OBJ SCARDS=INPUT T=1
```

```
Program GainCorr(input,output);
type vectorsize=array (.1..200.) of real;
    matrixsize=array(.1..30,1..30.) of real;
    matrixsize2=array(.1..4,1..30.) of real;
var Amatrix:matrixsize;
   Cmatrix:matrixsize2;
   gain,dgain,gain_phase,delta,diff,
   vector,sum,divisor,
   dvector_new,temp,mult,star_phase,star,
   corr,zeta:vectorsize;
   gain_in,gain_phase_in,comparison,
   HJD_Corr,hold,star_phase_in,star_in:real;
   counter,cnt,i,j,k:integer;
{ }
```

B.2.1. PROCEDURE READIN

Input procedure. The first entries consist of phase and gain values followed by data.

```
procedure readin;
begin
 counter:=1;
 cnt:=1;
 repeat
```

```
    readln(gain_phase_in,gain_in);
    gain_phase(.counter.):=gain_phase_in-HJD_Corr;
    gain(.counter.):=gain_in;
    counter:=counter+1;
  until (gain_phase_in  1);
  counter:=counter-1;
  repeat
    readln(star_phase_in,star_in);
    star_phase(.cnt.):=star_phase_in;
    star(.cnt.):=star_in;
    cnt:=cnt+1;
  until eof;
  cnt:=cnt-1
end;
{ }
```

B.2.2. PROCEDURE GAINDELTA

A simple calculation of successive differences in phase.

```
procedure gaindelta;
var i:integer;
begin
  for i:=1 to (counter-1)
    do begin
      delta(.i.):=gain_phase(.i+1.)-gain_phase(.i.)
    end
end;
{ }
```

B.2.3. PROCEDURE DIVISORDIFF

A simple calculation of derivatives.

```
procedure divisordiff;
var i:integer;
begin
 for i:=1 to (counter-1)
   do begin
     diff(.i.):=(gain(.i+1.)-gain(.i.))/delta(.i.)
   end
end;
{  }
```

B.2.4. PROCEDURE GAIN_S

Duplicate of DIVISORDIFF.

```
procedure gaind_s;
begin
 dgain(.1.):=diff(.1.);
 dgain(.counter-1.):=diff(.counter-1.);
end;
{  }
```

B.2.5. PROCEDURE FORM_MATRIX

Initialization of system of equations.

```
procedure form_matrix;
var i,j:integer;
begin
 for i:=1 to (counter-2)
   do begin
     for j:=1 to (counter-1)
do begin
```

```
 Amatrix(.i,j.):=0
end
   end
end;
{    }
```

B.2.6. PROCEDURE FILL_MATRIX

This procedure fills the matrix representing the data and relevant derivatives.

```
procedure fill_matrix;
var i,j,entryone,entrytwo,entrythree:integer;
begin
 Amatrix(.1,1.):=2*(delta(.1.)+delta(.2.));
 Amatrix(.1,2.):=delta(.1.);
 Amatrix(.1,counter-1.):=3*((delta(.1.)*diff(.2.))+
(delta(.2.)*diff(.1.)))-(delta(.2.)*diff(.1.));
 Amatrix(.counter-2,counter-3.):=delta(.counter-1.);
 Amatrix(.counter-2,counter-2.):=2*(delta(.counter-2.)+
delta(.counter-1.));
 Amatrix(.counter-2,counter-1.):=3*((delta(.counter-2.)*
diff(.counter-1.))+(delta(.counter-1.)*diff(.counter-2.)))-
(delta(.counter-2.)*diff(.counter-1.));
 entryone:=1;
 entrytwo:=2;
 entrythree:=3;
 for i:=2 to (counter-3)
  do begin
    Amatrix(.i,entryone.):=delta(.i+1.);
    Amatrix(.i,entrytwo.):=2*(delta(.i.)+delta(.i+1.));
    Amatrix(.i,entrythree.):=delta(.i.);
    Amatrix(.i,counter-1.):=3*((delta(.i.)*diff(.i+1.))+
    (delta(.i+1.)*diff(.i.)));
     entryone:=entryone+1;
     entrytwo:=entrytwo+1;
     entrythree:=entrythree+1
    end
end;
{   }
```

B.2.7. PROCEDURE INITIALIZE_SOLN

Formation of solution vector.

```
procedure initialize_soln;
var i,j:integer;
begin
 for i:=1 to (counter-2)
   do begin
    dvector(.i.):=0
   end
end;
{   }
```

B.2.8. PROCEDURE J_METHOD

This procedure solves the system of equations using a Jacobi iteration scheme (cf. Anton, p.321-327).

```
procedure J_Method;
var i,j,entryone,entrytwo,entrythree:integer;
begin
 entryone:=1;
 entrytwo:=2;
 entrythree:=3;
 dvector_new(.1.):=(1/Amatrix(.1,1.))*(Amatrix(.1,counter-1.)-
   (Amatrix(.1,2.)*dvector(.2.)));
 dvector_new(.counter-2.):=(1/Amatrix(.counter-2,counter-2.))*
   (Amatrix(.counter-2,counter-1.)-
   (Amatrix(.counter-2,counter-3.)*
    dvector(.counter-3.)));
 for i:=2 to (counter-3)
    do begin
      dvector_new(.i.):=(1/Amatrix(.i,entrytwo.))*
(Amatrix(.i,counter-1.)-
(Amatrix(.i,entryone.)*
 dvector(.entryone.))-
(Amatrix(.i,entrythree.)*
 dvector(.entrythree.)));
      entryone:=entryone+1;
      entrytwo:=entrytwo+1;
      entrythree:=entrythree+1
    end;
 comparison:=0;
 for i:=1 to (counter-2)
    do begin
      comparison:=comparison+abs(dvector_new(.i.)-dvector(.i.))
    end;
 if comparison1E-4 then
    begin
      for i:=1 to (counter-2)
 do begin
  dvector(.i.):=dvector_new(.i.)
 end;
      J_Method
    end
end;
```

```
{ }
```

B.2.9. PROCEDURE FORM_CMATRIX

This procedure forms the matrix of cubic spline coefficients used in the gain correction.

```
procedure form_cmatrix;
var i,j:integer;
begin
 Cmatrix(.3,1.):=diff(.1.);
 temp(.1.):=diff(.1.);
 temp(.counter.):=diff(.counter-1.);
 for j:=2 to (counter-1)
   do begin
    temp(.j.):=dvector_new(.j-1.)
   end;
 for j:=2 to (counter-1)
   do begin
     Cmatrix(.3,j.):=dvector_new(.j-1.)
   end;
 for j:=1 to (counter-1)
   do begin
     Cmatrix(.1,j.):=(temp(.j+1.)+
   temp(.j.)-
   2*diff(.j.))/sqr(delta(.j.));
     Cmatrix(.2,j.):=((diff(.j.)-
    temp(.j.))/delta(.j.))-
    (Cmatrix(.1,j.)*delta(.j.));
     Cmatrix(.4,j.):=gain(.j.)
   end
end;
{   }
```

B.2.10. PROCEDURE REMOVE_GAIN

This procedure locates the appropriate spline for the given data phase and calculates the gain correction.

```
procedure removegain;
var i,j,k,l:integer;
begin
 for i:=1 to cnt
   do begin
     corr(.i.):=0;
     if (star_phase(.i.)gain_phase(.2.))
 then begin
    zeta(.i.):=star_phase(.i.)-gain_phase(.1.);
    mult(.1.):=zeta(.i.)*zeta(.i.)*zeta(.i.);
    mult(.2.):=zeta(.i.)*zeta(.i.);
    mult(.3.):=zeta(.i.);
    mult(.4.):=1;
    for j:=1 to 4
do begin
 corr(.i.):=corr(.i.)+cmatrix(.j,1.)*
    mult(.j.);
end
 end
      else
      if (star_phase(.i.)gain_phase(.counter-1.))
 then begin
    zeta(.i.):=star_phase(.i.)-gain_phase(.counter-1.);
    mult(.1.):=zeta(.i.)*zeta(.i.)*zeta(.i.);
    mult(.2.):=zeta(.i.)*zeta(.i.);
    mult(.3.):=zeta(.i.);
    mult(.4.):=1;
```

```
       for j:=1 to 4
  do begin
    corr(.i.):=corr(.i.)+cmatrix(.j,counter-1.)*
       mult(.j.)
  end
  end
        else
        begin
  for l:=2 to (counter-2)
     do begin
       if (star_phase(.i.)gain_phase(.l.))and
  (star_phase(.i.)gain_phase(.l+1.))
    then begin
  zeta(.i.):=star_phase(.i.)-gain_phase(.l.);
  mult(.1.):=zeta(.i.)*zeta(.i.)*zeta(.i.);
  mult(.2.):=zeta(.i.)*zeta(.i.);
  mult(.3.):=zeta(.i.);
  mult(.4.):=1;
  for j:=1 to 4
     do begin
       corr(.i.):=corr(.i.)+cmatrix(.j,l.)*
  mult(.j.);
     end
      end
    end
end
  end
end;
{ }
```

B.2.11. PROCEDURE PUTOUT4

A simple output procedure.

```
procedure putout4;
var i:integer;
begin
 for i:=1 to cnt
   do begin
    writeln(star_phase(.i.):8:4,(star(.i.)-corr(.i.)):8:4)
   end
end;
{ start main }
begin
 readin;
 gaindelta;
 divisordiff;
 gaind_s;
 form_matrix;
 fill_matrix;
 initialize_soln;
 J_Method;
 form_cmatrix;
 removegain;
 putout4
end.
```

B.3. PROGRAM SORT_AVE

This program sorts and bins input data. The bin size may be varied to allow for different weighting schemes.

Compilation of the program is achieved by

```
$RUN *PASCALVS SCARDS=SORT.AVE SPUNCH=SAVE.OBJ T=1
```

Followed by

```
$RUN *SAVE.OBJ SCARDS=SAVE.OBJ SCARDS=INDATA
SPUNCH=OUTDATA T=1
```

Note : Procedures SelectSort and Switch are taken in their entirety from Oh!Pascal! (Cooper and Clancy, p.311).

```
Program Sort_Ave(input,output);

type stardata = record

mag : real;

phase : real

end;

   recordarray = array(.1..5000.) of stardata;

var  arraylimit,arraysize : integer;

   workingdata : recordarray;

{   }
```

B.3.1. PROCEDURE BINER

This procedure calculates the average phase, data and number of entries in a given bin.

```
Procedure Biner(var subarray : recordarray;

var sublength : integer;

var binsize,top,bottom : real);

var j : integer;

   binstart,binend,magadder,phaseadder,dividor : real;

   found : boolean;

begin
```

```
  binstart := bottom;
  binend := bottom + binsize;
  magadder := 0.0000;
  dividor := 0.0;
  found := false;
 while (binend  top)
 do begin
  for j :=1 to sublength
  do  begin
    if (subarray(.j.).phase  binend) and
(subarray(.j.).phase  binstart) then
 begin
  magadder := magadder + subarray(.j.).mag;
  dividor := dividor + 1.0;
  found := true
 end
  end;
    if (found = true) then
      begin
 writeln(' ',(binstart+(binend-binstart)/2):6:4,' ',
     (magadder/dividor):6:4,' ',trunc(dividor));
      end;
  magadder := 0.0000;
  phaseadder := 0.0000;
  found := false;
  dividor := 0.0;
  binstart := binstart + binsize;
  binend := binend + binsize;
 end;
end;
```

```
{   }
```

B.3.2. PROCEDURE AVERAGE

This procedure determines the location and size of bins for a data set.

```
Procedure Average(var arraylength : integer;
  var averageArray : Recordarray);
var finalarray : recordarray;
    lower,upper,size : real;
    location,l,length,initial,j  : integer;
    activated : boolean;
begin
  length := 0;initial := 1;activated := false;
  for location := 1 to 5
  do begin
  if (location = 1)
    then begin
  lower := 0.0000;
  upper := 0.1800;
  size := 0.0100
end;
  if (location = 2) then begin
  lower := 0.1800;
  upper := 0.3200;
  size := 0.0050
end;
  if (location = 3) then begin
  lower := 0.3200;
  upper := 0.6800;
  size := 0.0100
end;
```

```
 if (location = 4) then begin
 lower := 0.6800;
 upper := 0.8200;
 size := 0.0050
end;
 if (location = 5) then begin
 lower := 0.8200;
 upper := 0.9999;
 size := 0.0100
end;
j := 1;
 for l :=  arraylength downto 1
   do begin
  if (averagearray(.l.).phase  upper)  and (averagearrav(.l.).phase   lower)
  then begin
 finalarray(.j.).phase := averagearray(.l.).phase;
 finalarray(.j.).mag := averagearray(.l.).mag;
 length := length + 1;
 j := j + 1;
 activated := true
    end
  end;
    if activated
 then begin
biner(finalarray,length,size,upper,lower);
length := 0;
activated := false
 end;
 end
end;
```

```
{}
```

B.3.3. PROCEDURE SWITCH

A simple swapping routine used in sorting the data.

```
procedure Switch(var first,second : stardata);
var temporary : stardata;
begin
 temporary :=  first;
 first := second;
 second := temporary
end;
{            }
```

B.3.4. PROCEDURE SELECTSORT

This procedure performs a simple selection sort of the data by phase.

```
procedure SelectSort(var orderedarray : recordarray;
    var arraylimit : integer);
var first,largest,comparisons,switches : integer;
   current : 1..5000;
begin
 comparisons := 0;
 switches := 0;
 for first := 1 to arraylimit-1
   do begin
largest := first;
for current := first to arraylimit
  do begin
if (orderedarray(.current.).phase
   orderedarray(.largest.).phase)
then largest := current
```

```
  end;
Switch(orderedarray(.Largest.),
     orderedarray(.First.))

   end;
 Average(arraylimit,orderedarray)
end;
{
   Start of Main
}
begin
 arraysize := 1;
 while not eof
      do begin
   readln(workingdata(.arraysize.).phase,
  workingdata(.arraysize.).mag);
   arraysize := arraysize + 1
      end;
      arraysize := arraysize - 1;
      SelectSort(workingdata,arraysize)
end.
```